THE UNIVERSITY OF HONG KONG

ICMA'97
April 28-30, 1997
Hong Kong

INTERNATIONAL CONFERENCE ON MANUFACTURING AUTOMATION

PROCEEDINGS
Volume Two

EDITORS: S.T. TAN, T.N. WONG and I. GIBSON

PROCEEDINGS

OF

THE INTERNATIONAL CONFERENCE

ON

MANUFACTURING AUTOMATION

ICMA'97

April 28-30, 1997
Hong Kong

Editors: S.T. Tan,T.N. Wong and I. Gibson
The University of Hong Kong

VOLUME TWO

Organized by

Sponsored by

K.C. Wong Education Foundation

Leeport (Holdings) Ltd.

Department of Manufacturing Engineering,
City University of Hong Kong

Department of Mechanical & Automation Engineering,
The Chinese University of Hong Kong

Department of Mechanical Engineering,
The University of Hong Kong

William M.W. Mong Engineering Research Fund,
Faculty of Engineering, The University of Hong Kong

Endorsed by the Rapid Prototyping Association of the Society of Manufacturing Engineers

For the complete set of two volumes, ISBN 962-85138-1-8
For volume 1, ISBN 962-85138-2-6
For volume 2, ISBN 962-85138-3-4

Foreword

I am delighted to welcome you to the International Conference on Manufacturing Automation (ICMA '97) organized by the Department of Mechanical Engineering, The University of Hong Kong

In the same spirit as the first ICMA held in 1992, this conference aims to bring together leading experts in the area of manufacturing automation with a view to providing a forum for exchange of ideas and research findings, and to fostering a better link between industry and academic research The programme contains 4 keynote lectures and a collection of some 180 contributions from over 25 countries, covering latest advances in a wide range of topics of manufacturing automation. I hope that your expectations of the Conference will be fulfilled and wish you a fruitful and enjoyable week.

The Organizing Committee wish to acknowledge the very generous support from the sponsors of this event. Thanks are due to all members of the International Scientific Committee and keynote speakers for helping to make this event a rich and truly international one. The assistance of all session chairmen, student helpers and secretaries is gratefully acknowledged The encouragement and facilities provided by the Head of the Department of Mechanical Engineering, The University of Hong Kong are most appreciated.

S .T. Tan
Chairman
Organizing Committee

April 1997

Committees

LOCAL ORGANIZING COMMITTEE

S.T. Tan (Chairman)	(HKU)
I. Gibson (Deputy Chairman)	(HKU)
T.N. Wong (Deputy Chairman)	(HKU)
K.W. Chan	(HKU)
Y.H. Chen	(HKU)
S.H. Choi	(HKU)
K.C. Hui	(CUHK)
C.L. Li	(CityUHK)
W.S. Sze	(HKU)
W. Wang	(HKU)
K.M. Yu	(HKPU)

INTERNATIONAL SCIENTIFIC COMMITTEE

A. Bowyer	(UK)
K. Case	(UK)
Y.K. Chan	(HK)
T.C. Chang	(USA)
N.N.Z. Gindy	(UK)
F.J.A.M. van Houten	(Netherlands)
T.L. Kunii	(Japan)
Y.T. Lee	(Singapore)
J.D. Liang	(Canada)
G.C.I. Lin	(Australia)
H. Marcus	(USA)
T. Medland	(UK)
A. Middleditch	(UK)
B. Mieritz	(Denmark)
T. Nakagawa	(Japan)
A.Y.C. Nee	(Singapore)
Q.S. Peng	(China)
T. Pintat	(Germany)
A.A.G. Requicha	(USA)
H.B. Voelcker	(USA)
J.Y. Wang	(Singapore)
Y.G. Wang	(China)
T. Wohlers	(USA)
J.R. Woodwark	(UK)
Y. Yan	(China)
M. Yeung	(Canada)
M.F. Yuen	(HK)
S. Zhang	(China)
J. Zhou	(China)

TABLE OF CONTENTS

IDENTIFYING AND EXPLAINING INFEASIBLE ASSEMBLY OPERATIONS

Maria João Abrantes and Simon D. Hill
Department of Digital Systems, Monash University
Clayton, VICTORIA 3168, Australia,
mja@dgs.monash.edu.au.

ABSTRACT

This paper presents a knowledge-based method for identifying infeasible assembly operations in the context of assembly sequence planning. In addition the reasons for the infeasibility are identified and these can guide product designers towards the necessary design changes to make a preferred assembly sequence feasible, or guarantee that an assembly sequence includes a specific partial assembly state. This capability increases the usefulness of assembly sequence planners as concurrent engineering tools by contributing to a closer integration between product design and assembly sequence planning.

In the identification of infeasible assembly operations the concept of minimal infeasible sets of connections is introduced and two issues are addressed: the knowledge-based generation of these minimal sets, and their suitability to form a basis for a concise explanation of the causes of assembly operation infeasibility. Two simple examples illustrate practical applications of the method.

KEYWORDS

Assembly Sequence Planning, Knowledge-Based Systems, Concurrent Engineering

1. INTRODUCTION

During the last decade significant advances have been made in the automation of the assembly sequence planning process. The main motivation behind the development of automatic assembly sequence planners is the great usefulness of such a tool in a concurrent engineering (CE) approach to product development. The choice of the assembly sequence of a product impacts on, and is also constrained by, many other aspects of the development cycle of a product, such as, product design, product testing and repair strategies, and assembly system design. Automatic assembly sequence planning tools allow fast and reliable generation of assembly sequences and, therefore, facilitate the evaluation of the mutual interactions between different aspects of product development. Furthermore, some tools integrate assembly sequence planning and product design with a view to optimizing assembly sequence and design in Design for Assembly (DFA) terms [1, 2].

However, an aspect of the interaction between product design and assembly sequence planning that has not received much attention is that the infeasibility of an assembly sequence may, especially in a CE environment, give rise to product redesign decisions. For example, if a product's design does not allow a preferred assembly sequence in terms of assembly system design, it may fall to the designer to identify the design aspects that preclude the sequence and devise design changes needed to make the sequence available. The knowledge-based method discussed in this paper performs the first of these tasks and can guide designers in the second task. The automatic identification of the causes for the infeasibility of assembly operations complements existing product design and assembly sequence planning integration methods, hence increasing the usefulness of assembly sequence planners as concurrent engineering tools.

The approach used here is centred on the identification of minimal sets of connections that cannot be established in a single assembly operation. The advantage of identifying these minimal infeasible sets of connections is twofold: i) these sets constitute a convenient basis for a concise operation infeasibility explanation facility, and ii) they provide a means of increasing planning efficiency by maximizing the logical inference of assembly operation feasibility. The methods described in this paper increase the efficiency of the system reported in [3, 4].

2. TERMINOLOGY AND ASSUMPTIONS

An assembly is considered to be a set of interconnected parts forming a stable unit [5]. Each part is a solid object, but it may exhibit some compliance if the part's deformation is inherent to the establishment of a connection, such as in the case of snap-fit or press-fit connections.

Connections in an assembly can be real or virtual connections. A real connection exists between any pair of parts with at least one surface contact between them [5]. A virtual connection is a logical connection between a pair of parts with no physical contacts, or with plane-point, plane-edge, edge-edge and edge-point contacts, such that a blocking relationship exists between the parts. Unlike real connections, virtual connections may exist between parts that are not adjacent in an assembly.

A subassembly is a non-empty subset of the set of parts that constitute an assembly that either has only one part, or is such that every part has at least one real connection with another part in the subset [5]. It is assumed that (i) when a subassembly is formed all connections between its parts are established and will remain established throughout the assembly process and (ii) an assembly operation joins exactly two subassemblies [5] by moving one subassembly towards another that is considered to be stationary.

To reduce the dimensionality of the problem to manageable size, two simplifying assumptions are made regarding the establishment of connections. Firstly, a connection's establishment motion may consist of a single translation or a simultaneous single translation and rotation (as in the fastening of a screw) of a moving part to its goal position in the assembly. Secondly, there are six potential assembly directions, each respectively along the positive or negative axes of a global Cartesian reference coordinate frame (the Assembly Reference Frame).

Given this, a set of potential connection establishment directions, or assembly directions, can be defined by $D = \{+x, -x, +y, -y, +z, -z\}$, where the symbols + and - define the direction of the establishment motion, or translational component of the establishment motion, with respect to the relevant axis of Assembly Reference Frame. As such, this paper deals with the group of assemblies whose connections can all be established along orthogonal assembly directions.

It is assumed that in an assembly operation the moving subassembly's motion begins at an insertion position, that is, neither the initial position of the subassembly (e.g. in a part feeder, magazine, or subassembly fixture) nor the trajectory of the subassembly between this initial position and the insertion position are considered.

3. AN OVERVIEW OF THE KNOWLEDGE-BASED APPROACH

As mentioned earlier, a knowledge-based system may be used to automatically identify infeasibilities and complement existing product design and assembly sequence planning integration methods. Irrespective of the assembly sequence planning approach used some basic assembly knowledge is required. The knowledge-based approach described exploits this basic assembly knowledge and has a number of advantages. Firstly, it attempts to make use of the information available efficiently. Secondly, it may be used to reduce the use of computationally intensive geometric reasoning in CAD-based systems. Thirdly, it may reduce and simplify the level of user interaction in non-geometric interactive systems. Finally, and perhaps most importantly, it facilitates the use and understanding of the interaction between product design and feasible assembly sequence options.

The system's inputs consist of an initial assembly model and any eventual user-defined assembly constraints, such as the type of required sequences (linear or non-linear), the specification of a base-part, and the connections establishment precedence constraints. The initial assembly model contains only the information that cannot be inferred, which must be provided anyway. For example, the only virtual connections in the initial model are virtual connections between parts which are adjacent in the assembly.

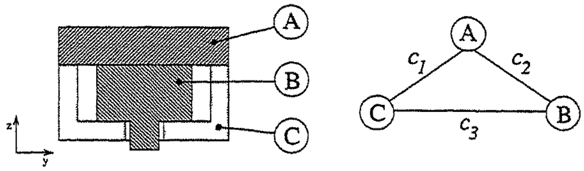


Figure 1: A simple assembly and its graph of connections

The system's static assembly knowledge enables subsequent simplifications[1] and enhancements of the initial model, which are necessary for the correct and efficient generation of assembly sequences. For some assemblies the enhancement of the initial model (which consists basically of the logical inference of virtual, or potential virtual connections) is completely automatic, while for other assemblies the system may prompt the user for some extra information, or geometric algorithms in the case of CAD-based systems.

Because the initial assembly model contains only essential data, it can be easily created manually from a sketch of the product if this sketch is exact in terms of topological relationships between parts. As such the system can be used even before CAD drawings or fully-dimensioned engineering drawings exist and can be used at the conceptual design stage of a product, when the design team has the greatest opportunity to explore alternatives and identify the best product and manufacturing process [6].

The assembly knowledge, in the form of production rules, is represented in a system knowledge base which is partitioned into several modules for efficiency reasons. This modular construction allows the system to focus on one or more modules at a time during execution, in which system access is restricted to constructs (certain rules, classes, etc.) and the portion of the working memory (certain facts and instances) which are relevant to the current module or modules.

4. IDENTIFYING INFEASIBLE OPERATIONS

Precedence constraints are widely used in assembly sequence planning [7, 8, 9, 10], either as a starting point for the generation of feasible assembly sequences, or within this generation process for the logical evaluation of assembly operations, thus avoiding more computationally expensive direct evaluation tests and increasing planning efficiency.

In both approaches the infeasibility of assembly operations is viewed in terms of a precedence constraint violation. For example, in Figure 1, the operation joining $\{B\}$ to $\{A, C\}$ is infeasible because it violates a precedence constraint of the type: *the establishment of connections c_2 or c_3 must precede the establishment of connection c_1*. However, the infeasibility of an assembly operation can alternatively be viewed in terms of the infeasibility of establishing the set of connections that must be established in that operation. That is, in Figure 1 joining $\{B\}$ to $\{A, C\}$ is infeasible because *it is not possible to establish connections c_2 and c_3 in a single assembly operation.*

Before discussing the rule-based identification of minimal infeasible sets of connections the following three definitions are required:

Definition 1: Minimal Infeasible Set of Connections A set of n connections $C_n = \{c_1, c_2, ..., c_n\}$ is a minimal infeasible set of connections *iff* it cannot be established in a single assembly operation, and any set of m connections, such that $C_m \subset C_n$, can be established in a single assembly operation.

[1]These include deletion of redundant connections from the graph of real connections and temporary grouping of parts into *supernodes* (a fundamental procedure regarding the efficiency of cutset-based systems).

Let $c_{p_ip_j}$ represent the connection between parts p_i and p_j, where p_i and p_j are respectively the moving and the stationary part, and let d be a potential assembly direction, that is $d \in D$.

Definition 2: Correct Connection Establishment Connection $c_{p_ip_j}$ is *correctly established* in direction d *iff* there is a connection's establishment motion for which p_i does not interfere with p_j, or any interference between the two parts is strictly inherent to the mechanics of the establishment of the connection.

Definition 3: Connection Establishment Directions The set of establishment directions of connection $c_{p_ip_j}$ is the set of directions $E_{c_{p_ip_j}} \subset D$, for which the connection is correctly established. For example, in Figure 1 connection c_{AB} has five possible establishment directions, i.e. $E_{c_{AB}} = \{+x, -x, +y, -y, -z\}$, whilst connection c_{BC} has only one possible establishment direction, $E_{c_{BC}} = \{-z\}$.

A sufficient condition for the infeasibility of the establishment of a set of n connections $C_n = \{c_1, c_2, \ldots, c_n\}$ is the non-existence of an establishment direction common to all connections, that is $E_{c_1} \cap E_{c_2} \cap \ldots \cap E_{c_n} = \varnothing$ (**condition 1**). Moreover, C_n is a minimal infeasible set of connections if for any set of connections $C_m = \{c_{i_1}, c_{i_2}, \ldots, c_{i_m}\}$, such that $C_m \subset C_n$, $E_{c_{i_1}} \cap E_{c_{i_2}} \cap \ldots \cap E_{c_{i_m}} \neq \varnothing$ (**condition 2**). To efficiently find the subsets of the set of connections to be established in an assembly operation, that observe conditions 1 and 2 above, connections were classified into connection types according to their sets of establishment directions. These latter sets are subsets of all assembly directions D, thus, the set of all (admissible) connection types, T, is defined as: $T = \{t : \mathbb{P}D \mid 0 < \#t < 6\}$. Also, since the axes of the Assembly Reference Frame are labelled arbitrarily (as x, y, or z), sets of connection types can be considered to be equivalent under relabelling of axes. Thus, the admissible connection types T can be partitioned into eight equivalence classes (EC's) according to axes relabelling. Let $[T]$ denote the EC's of admissible types, for example $[\{+x, -x\}] = \{\{+x, -x\}, \{+y, -y\}, \{+z, -z\}\}$, and $[\{+x, +y\}] = \{\{+x, +y\}, \{+x, +z\}, \{+y, +z\}, \{-x, -y\}, \{-x, -z\}, \{-y, -z\}, \{+x, -y\}, \{+x, -z\}, \{+y, -z\}, \{-x, +y\}, \{-x, +z\}, \{-y, +z\}\}$. Furthermore, EC's are ranked according to their cardinalities (the ranking between EC's of equal cardinality being decided arbitrarily).

Conditions 1 and 2 can be easily evaluated by a conveniently structured set of rules which verifies that the types of the connections to be established in an operation observe specific combinations of their equivalence classes. These rules are structured so that EC's are always combined with EC's of equal or higher rank, and since $0 < \#t < 6$, the number n of EC's combined is such that $1 < n < 6$. For example, in the rules that identify minimal infeasible sets of two connections, EC $[\{+x\}]$ is combined with any EC, whilst EC $[\{+x, +y, +z\}]$ is only combined with itself because for any EC of higher rank the two connections will always have at least one common establishment direction. This rule structuring avoids more than one rule firing for each minimal infeasible set of connections.

For operations involving only the establishment of translational connections, condition 1 is a necessary and sufficient condition for the operations infeasibility. For threaded connections, however, condition 1 is sufficient but not necessary. A specific set of heuristic rules tests a series of other sufficient conditions for the infeasibility of operations involving threaded connections. Examples of these sufficient conditions are operations involving the establishment of two threaded connections with: non-concentric threads, or different types of thread (e. g. number of threads per inch, fastening rotational direction), and operations involving a threaded connection and any other connection that would imply an obstruction of the rotational component of the establishment motion of the threaded part (or of any part in the subassembly of a threaded part).

Since a connection is established by moving one part towards a stationary part, a connection c is conveniently represented by a 3-tuple (m, s, t), in which m is the moving part, s is the stationary part, and t is the *connection type*, that is $t \in T$. Interchanging the roles of the movable and stationary parts, the establishment directions are opposite to the original ones. Let C designate the class of all connections. A dual connection is thus defined as: $- : C \to C$, where $-(m,s,t) = (s,m,-t)$. Clearly, any connection c and its dual belong to the same connection type EC.

Dual connections are derived so that for the connections established in an operation the first elements of the 3-tuples representations are parts belonging to one of the subassemblies and the second elements are parts belonging to the second subassembly. For the assembly in Figure 1, the set of connections established in the operation joining subassembly $\{A\}$ (moving) to $\{B,C\}$ (stationary) is $C = \{(A,B,\{+x,-x,+y,-y,-z\}),](A,C,\{+x,-x,+y,-y,-z\})\}$, and in the operation joining $\{B\}$ to $\{A,C\}$ is $C = \{(B,A,\{-x,+x,-y,+y,+z\}),(B,C,\{-z\})\}$.

Finally, minimal infeasible sets of connections are identified for interference between parts, interference between parts and assembly tools (currently with limited capabilities), and violation of user-defined constraints.

5. EXPLAINING ASSEMBLY OPERATION INFEASIBILITY

To illustrate the way in which the system produces explanations, consider the assembly shown in Figure 2. This assembly consists of a container, B and a lid, A. Parts C and D are inside B and part E holds D in place.

Example 1: Consider that for technical reasons subassembly $\{D,E\}$ must be constructed separately.

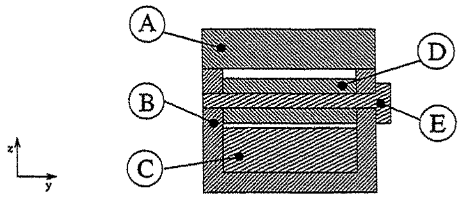


Figure 2: A simple assembly example

The system provides the following infeasibility explanation in the evaluation of the operation joining subassemblies $\{D,E\}$ and $\{B,C\}$:

```
Subassemblies are:
1-1 composed by: D E
1-2 composed by: B C
Operation joining (D E) and (B) is infeasible.
REASON => Connections between D and B (-z) and between E and B (-y)
      => are two insertions along orthogonal directions.
```

The explanation signals infeasibility because parts D and E must both be inserted into part B, but do not share a common establishment direction. Design changes to make this operation feasible will be needed to the types of these connections so that a common establishment direction is possible. One solution would be to make the establishment of both connections feasible in the $-z$ direction by modifying parts A and B as shown in Figure 3(a).

Example 2: Consider that product quality control requires subassembly $\{B,D,E\}$ to be tested before part C is added to it, for the assembly in Figure 2. The system provides the following explanations in the evaluation of the operation joining $\{C\}$ to $\{B,D,E\}$:

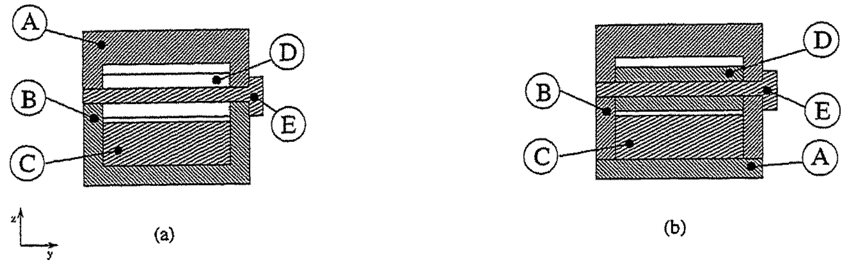


Figure 3: Design alternatives for the assembly of Figure 2

```
Subassemblies are:
1-1 composed by: C
1-2 composed by: B D E
Operation joining (C) and (B D) is infeasible.
REASON => C cannot be joined to B (-z) because C collides with D.
Operation joining (C) and (B E) is infeasible.
REASON => C cannot be joined to B (-z) because C collides with E.
```

As can be seen C can only be assembled to B in the $-z$ direction, but in this assembly direction C is obstructed by D and E. An obvious solution would be to modify the connection between parts C and B, c_{CB}, to allow an establishment direction other then $-z$, which could be easily achieved by changing the position of lid A as shown in Figure 3(b).

6. CONCLUSIONS

This paper has described a knowledge-based system for the generation of feasible assembly sequences and a new method based on minimal infeasible sets of connections for its operation. The approach allows concise explanations of assembly operation infeasibility, which provide advantages over other approaches particularly with regard to redesign issues.

7. REFERENCES

[1] G. J. Kim, S. Lee, and G. A. Bekey, 'Interleaving Assembly Planning and Design', IEEE Transactions on Robotics and Automation, 12(2):246-251, April 1996.

[2] W. Hsu, C. S. G. Lee and S. F. Su, 'Feedback Approach to Design for Assembly by Evaluation of Assembly Plan' Computer-Aided Design, 25(7):395-410, July 1993.

[3] M. J. Abrantes and S. D. Hill, 'Expert System Determination of Feasible Assembly Operations', Industrial and Engineering Applications of A. I. and Expert Systems: Proc. Eighth Int. Conf., G. Forshyth and M. Ali, pp. 181-187, Gordon and Breach, Melbourne, 1995.

[4] M. J. Abrantes and S. D. Hill Computer-Aided Planning of Mechanical Assembly Sequences, Technical Report 95-7, Department of Robotics and Digital Technology, Monash University, July 1995.

[5] L. S. Homem de Mello, Task Sequence Planning for Robotic Assembly, PhD thesis, Chapters 1 and 3, Carnegie-Mellon University, Pittsburgh, 1989.

[6] Henry W. Stoll, 'Simultaneous Engineering in the Conceptual Design Phase', Simultaneous Engineering Integrating Manufacturing and Design, C. W. Allen, pp. 165-175, SME, Dearborn Mich., 1990.

[7] A. Delchambre, Computer-Aided Assembly Planning, Chapter 5, Chapman and Hall, London, 1992.

[8] D. F. Baldwin, T. E. Abell, M-C M. Lui, T. L. De Fazio , and D. E. Whitney, 'An Integrated Computer Aid for Generating and Evaluating Assembly Sequences for Mechanical Products', IEEE Transactions on Robotics and Automation, 7(1):78–94, February 1991.

[9] V. N. Rajan and S. Y. Nof, 'Minimal Precedence Constraints for Integrated Assembly and Execution Planning', IEEE Transactions on Robotics and Automation, 12(2):175–186, April 1996.

[10] R. Wilson, On Geometric Assembly Planning, PhD thesis, Chapter 6, Stanford University, Sanford CA, 1992.

ARTIFICIAL NEURAL NETWORKS FOR PERFORMANCE ESTIMATION IN WOOD TURNING

V. Karri
Dept. of Mechanical Engineering
University of Tasmania, Hobart, Australia - 7001

ABSTRACT

A reliable performance estimates such as the forces, power, temperatures, tool life and vibrations help in selection of machine tools, cutting tool and develop optimization strategies. Traditional approaches such as mechanics of cutting approach and empirical approaches have been found to be either complex or often expensive to build reliable quantitative predictive models. Neural networks are sought to facilitate automatic control of cutting tools, make intelligent decisions and develop performance predictive models. In this paper, a neural network architecture is developed to estimate the force components in wood turning operations. The network to estimate all the three force components is trained over a comprehensive range of experimental conditions covering a range of tool-geometrical combinations.

KEYWORDS

Neural Networks, Manufacturing Processes, Intelligent Machines, Machining

1. INTRODUCTION

Turning is one of the largest metal cutting operations carried out in component manufacture around the world. Turning is a material removal process which is performed on a lathe to generate external cylindrical and conical surfaces. In this process, the workpiece is rotated into a longitudinally fed single-point tool. The interference that occur between the cutting tool and workpiece then generates an external cylindrical surface at the desired component diameter. Turning is one of the widely studied practical machining operations with several performance predictive models developed over the years. Most of the developed models were related to machining of metals and machining of wood has not been extensively studied due to the anisotropic properties of this material. The fundamental mechanics of cutting approach gives a better understanding of the deformation, geometry and the tool-workpiece interference as well as the equations for all the deformation parameters, friction in machining and force components [1-9]. The anisotropic nature of the wood makes this approach very tedious and often unapproachable [10].

It is essential to check if the qualitative trends of the effect of major process variables on the forces and power for wood are similar to that of metals. Wood is highly anisotropic in nature, with its tensile strength much greater in the direction parallel to the stem. In other words, the tensile strength is much greater along the orientation of the grains. This aspect is of vital importance for engineering designs involving wood applications and care must be taken to ensure that members are configured, aligned and loaded so as to obtain optimum results. The axis parallel to the tree stem or fibre (grain) is the longitudinal axis (L), the radial axis (R) is normal to the growth rings (perpendicular to the grain, in the radial direction). The tangential axis (T) is parallel to the annual rings and is perpendicular to both the radial and longitudinal axes as shown in fig.1. It has been found in the preliminary experimental investigation that when machined, using a single edged wedge tool, a continuous type chip is formed along the L-axis the chip formed, a discontinuous type chip is formed along the T-axis and along the radial direction the workpiece cracks ahead of the cutting tool. The steady state process and associated data acquisition requires a continuous chip formation and hence the machining was carried out along the L-axis only [11,12]. This has been successfully applied while testing the orthogonal cutting characteristics of machining Huon pine . A reliable mechanics of cutting model to cover all the three different axes of wood for performance prediction is a highly complex problem. By contract, the empirical approach to machining performance prediction relates machining performance features to numerous process variables with the aid of expensive and often prohibitive experimental testing and complex curve fitting techniques.

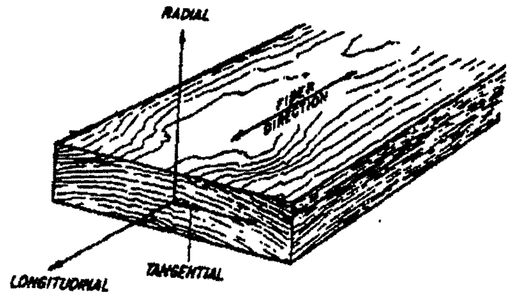


Fig.1 Three principal axes of wood

By contrast, using neural networks the performance prediction for machining operations can be carried out without associated complexity as other methods. It has already been shown by the chief investigator [13-15] that the performance predictions for single edged oblique cutting operations are possible using artificial neural networks and that there are significant advantages over other conventional methods. In this paper neural network architecture is developed for performance predictions in turning operations. The force components Ftang (tangential force in cylindrical turning), Frad (radial force component) and Ffeed (feed force component) [1,2,3,4] in turning operations will be predicted using artificial neural networks. A set of wood turning is carried out covering a range of tool geometrical conditions. This experimental data will be used as a part of training and testing the neural network developed.

2. DATA ACQUISITION AND EXPERIMENTAL RIG

The experimental rig developed involves the measurement of all the three force components during the cutting process. The concentric dynamometer with strain gauges is connected and mounted round on the cross slide of the lathe machine. The strain gauges were connected to the strain gauge amplifier for recording the force components. A data acquisition system was used together with a 486 IBM compatible computer for recording the force components. The cutting tools were mounted on the dynamometer for the measurement of three force components (Fp (power force), Fq (thrust force) and Fr (radial force)) in all three directions during cutting [1-4] as shown in the experimental set up in figure 2 below. The Huon pine wooden block was used as a work material and machined. In carrying out the turning tests five different cutting speeds, five different feed rates, five different depths of cut, three rake angles and three inclinations [2,4] were selected.

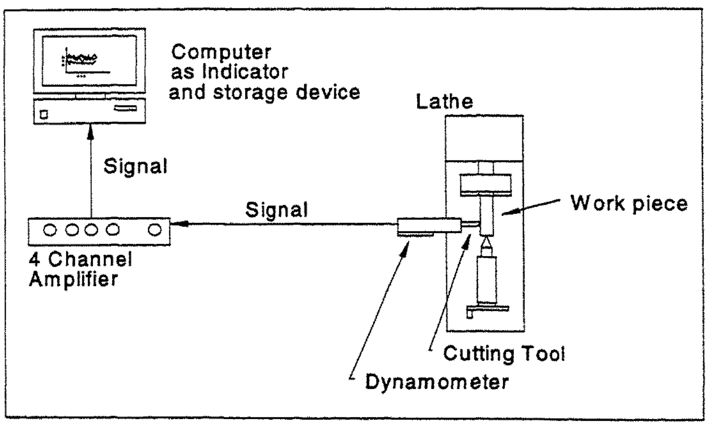


Fig.2 Experimental Set up

Before designing the architecture of the neural networks, the number of inputs needed are to be decided to predict all the three force components in turning operations. The number of inputs for neural network will be limited to 5 namely, the tool rake angle, inclination angle, feed, depth of cut and the velocity (rpm).

The target values for evaluating the output of the neural network will be provided from the experiments carried out using the above mentioned experimental rig. These cutting tests have been carried out on a lathe for cutting conditions covering 3 rake angles (0°,10°,20°), 3 inclination angles (0°,10°,20°), 5 feed rates (0.05, 0.1, 0.2, 0.3, 0.4 mm/rev.), 5 depths of cut (1.5mm) and five cutting speeds (35.2, 70.5, 132.4, 176.34, 244.16 m/min.) for each tool geometry so that a total of 205 cutting tests were used for training the network. The force components obtained through the data acquisition system will be used for training the network and a further 30 cutting tests were carried out to test the validity of the neural network architecture.

3. NEURAL NETWORK MODEL FOR PERFORMANCE ESTIMATION

The multi-layer perceptron [16] illustrated in fig.3 can be exploited to synthesise decision making with many desirable features. The architecture is made of an input layer, a hidden layer and an output layer. The hidden and output layers have processing units and interconnections called neurons and synapses respectively. Each interconnection has an associated connection strength or weight. The numbers of hidden layers and that of nodes (units) in each layer have to be decided very carefully, because the system cannot model the given information if it has too few hidden units; however, too many hidden units limit the network's ability to generalise the rule, so that the resulting model would not work well for new incoming data. Each processing unit first performs a weighted accumulation of the respective inputs and bias value, then passes the result through a sigmoidal activation function.

There are 5 inputs at the input layer stage namely the rake angle, inclination angle, feed, speed and depth of cut. The network architecture with the input values and the target values of force components supplied trains the network based on the back propagation algorithm. Back propagation can train multi-layered feed forward networks with differentiable transfer functions to perform function approximation, pattern association, and pattern classification. The back propagation follows gradient descent on the error surface to minimise the network error. Although adding momentum to back propagation can decrease training times and the probability that the network will get stuck in a shallow minimum in the error surface, in this particular application, no momentum is added.

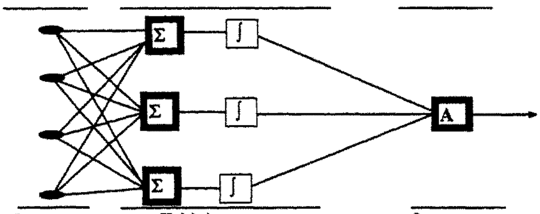


Fig.3 Neural Network Architecture for cutting tool selection

The training of the network is carried out for all the three forces. The target force components (as measured force components i.e. Ftang, Frad and Ffeed) were scaled between 0-1 and the training is carried out over 205 combinations of cutting conditions. The predicted Ftang., Ffeed and Frad components during the training stage had a mean percentage deviations of 2.5%, 1.6% and 1.5% respectively. The percentage deviation was defined as % dev (error) = (Target-Pred.)/Target * 100. The network was trained for 1.5 million pattern presentations which is sufficient in these cases to ensure that the network has reached a minimum. To check this networks were trained between 1 and 5 million pattern presentations and showed no significant differences in the mean error at the training stage (see figs. 4a,c,e). The testing conditions were well within the domain of the trained conditions. The multi-layer perceptron with back propagation program was run to check the validity of the neural network model for the testing stage. Figs.4b,d,f show the accuracy of the neural network model for all the three force components tested. It can be seen that Ftang,Frad and Ffeed are within ± 10 %. The tangential force, Ftang., is slightly over predicted with 4.5% deviation, the radial force, Frad is under predicted with 3.5% deviation and finally the feed force Ffeed is over predicted with a percentage deviation of 3.8%. The histograms drawn to check the scatter indicated that the percentage deviations were on both the sides of the mean so that there is no significant bias towards under or over predictions. This accuracy of prediction is particularly pleasing since only 5 inputs were used as inputs.

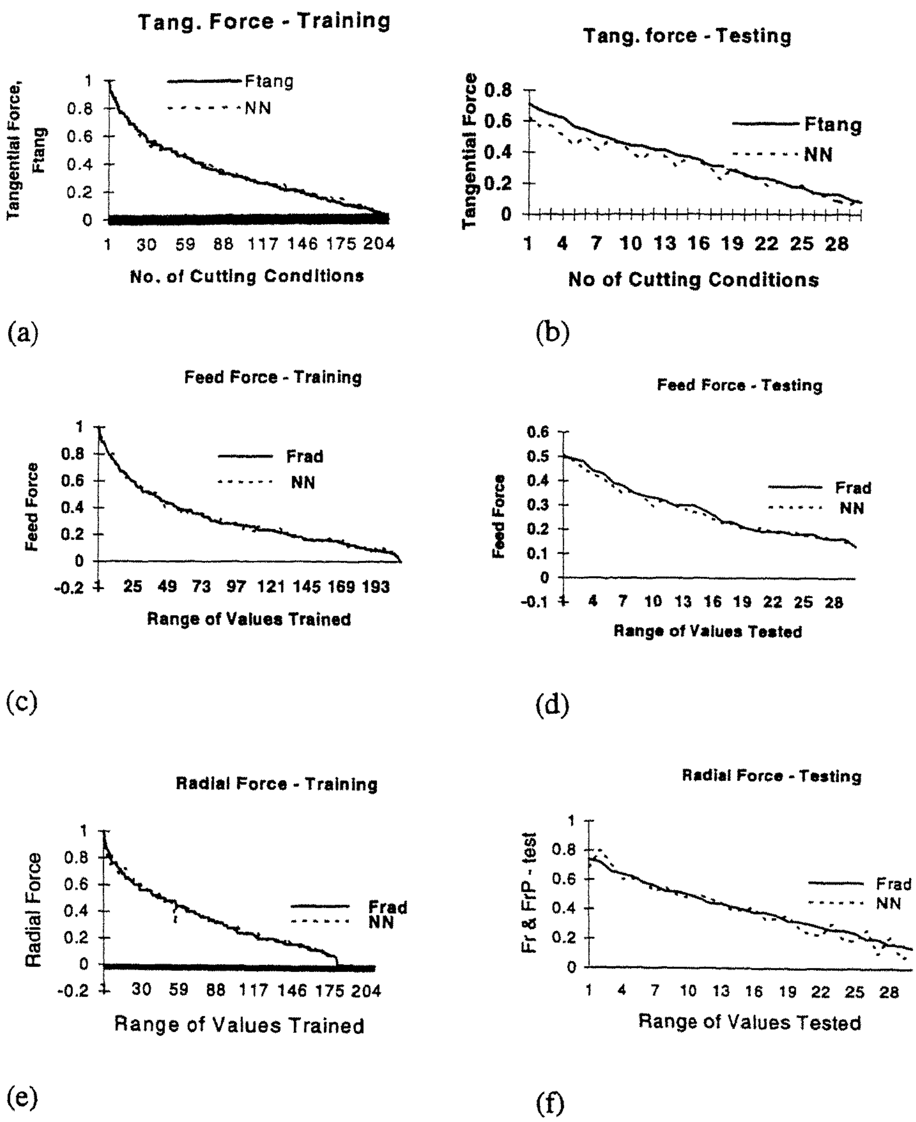


Fig.4. Force components at Training and Testing Stages

4. CONCLUSIONS

It has been discussed that from a predictive point of view, in turning operation, the mechanics of cutting approach has limitations and that a large number of inputs are required for quantitative predictions of the forces. Wood as an anisotropic material is discussed and the complication associated with the development of predictive models using conventional methods is highlighted. The extensive and often prohibitive experimentation involved in empirical approach often discourages to develop predictive models. By contrast, it has been shown that, using multi-layered perceptron with back propagation the neural network is trained to an accuracy of ± 3% error for all the three forces in turning operation. It has been shown that using the neural networks the inter-relationships between the process variables can also be trained so that a fewer number of inputs can be used to estimate the performance as against the other conventional methods.

The experiments have been carried out on Australian grown huon pine. A comprehensive set of data covering a range of tool-geometrical combination is gathered using sophisticated data acquisition methods. The developed neural network model is trained extensively over 200 cutting conditions encompassing a comprehensive range. The well trained network is tested for all the three force components and is found to be accurate to ± 5% error. The predictions obtained from the neural network model are quantitatively more accurate and reliable than other models available in the literature. The model has not shown any significant bias to over or under prediction.

5. REFERENCES

1. Armarego, E.J.A., " Practical Implications of Classical Thin Shear Zone Analysis", UNESCO/CIRP Seminar on Manuf.Tech., Singapore, (1982),p167.

2. Karri,V, Fundamental Studies of Rotary Tool Cutting Processes, Ph.D. Thesis, The University of Melbourne, (1991) .

3. Armarego, E.J.A., Brown, R.H., The Machining of Metals, Prentice Hall Inc., New-Jersey, 1969.

4. Uthaichaya, M, A Study of the Forces in Turning,M.Eng,. Thesis,The University of Melbourne,1974.

5. Armarego.E.J.A and Deshpande, N.P, "Computerised Predictive Cutting Models for Forces In End-Milling Including Eccentricity Effects", Annals CIRP, 1989,Vol.38/1,p.45.

6.Pramanik,D.,"Variables affecting drill Performance", Ph.D, Thesis,The Uni. Melbourne, 1988.

7. Karri,V., 1995, "Computer-Aided Predictive Cutting Model for Simulated Keyway Broaching Operation",3rd IASTED Int.Conf. Robotics and Manuf., Jun.14-17, Cancun, Mexico. pp.144, 1995.

8.Whitfield,R.C., Force Predictions in Machining, Ph.D., Thesis,The University of Melbourne,1986.

9. Karri, V., 'A Neural Network Controller in Automated Wood Machining'., IASTED Int. Conf. on Robotics and Manufacturing, Honolulu, USA. Aug.19-22, 247-251. (1996).

10. Freudenthal, A.M., The inelastic behaviour of engiering materials and structures, Wiley and Sons, Newyork., 1950.

11. Karri, V., "Orthogonal Cutting Characteristics of Huon Pine - An Experimental Investigation", 14th Int. Conf. on Mech. Struc. & Mat., Dec.,Hobart,p 664. 1995.

12. Karri, V., 1995, "Oblique Cutting Characteristics of Huon Pine", 14th Int. Conf. on Mech. Struc. & Mat., Dec.,Hobart,p 669, 1995.

13. Karri V, 'Performance Estimation in Wood Machining Using Artificial Neural Networks', Int. Conference on Advanced Manufacturing Processes Systems and Technologies, Bradford, England, pp.271. (1996).

14. Karri V, 'Performance Prediction models in Oblique Cutting Using Conventional and Neural Network Approaches - A Comparison', Int. Conference on Advanced Manufacturing Processes Systems and Technologies, Bradford, England, 281. (1996).

15. Madsen O (50%), Karri V (50%) 'Neural Networks to Estimate Travel Speed for Control of Root-Bead Welding', Int. Conference on Advanced Manufacturing Processes Systems and Technologies, Bradford, England, Mar. 122. (1996).

16.Rumelhart,D.E.,McCleland,J.L., "Parallel Distributed Processing", MIT press, Cambridge, 1986.

Multi-modelling : a help to rapid product development

Yvon Gardan
University of Metz
57045 Metz cedex, France

ABSTRACT :

Modelling is an important component of CAD/CAM systems. In powerful systems, it includes various kind of information : features, geometry, transformations ... We think that a multi-model scheme is well appropriate to represent the fact that the information is not the same in different stages of the CAD/CAM process. For example, features are not the same in design and manufacturing. We consider that the design model is a generic model, from which are derived different models (N.C., process management, finite elements ...), so called "applications" models. But this generic model must be considered in a concurrent engineering context and the process itself has to be described. In this paper, we describe the overall architecture of a multi-modelling base CAD system we are developing (REGAIN).

KEYWORDS :
CAD/CAM, multi-modelling, product data management

1. INTRODUCTION

Modelling is an important component of CAD/CAM systems. In powerful systems, it includes various kind of information : features, geometry, transformations ... We think that a multi-model scheme is well appropriate to represent the fact that the information are not the same in different stages of the CAD/CAM process. For example, features are not the same in design and manufacturing. We consider that the design model is a generic model, from which are derived different models (N.C., process management, finite elements ...), so called "applications" models.

The structure of the generic model is complex (and can be "distributed") and to improve performance and facilitate algorithms process, we extend the notion of "application" model to man-machine interface and visualisation. We define, for example, a product generic model as features representation, General Graphical Inputs and a constraints graph, deriving from this, dialogue models. Our aim is to give tools to implement application models : i.e. the end user must be able to design its CAD/CAM system, without computer science knowledge. In that, the interactive approach is favoured and the associated tools are the base of the system.

In this paper, we describe the overall architecture of a multi-modelling based CAD system we are developing (REGAIN)[1]. REGAIN has two main goals :

- to give a computer assisted CAD software environment ;
- to propose a development methodology.

An implementation of REGAIN is then a CAD software, using REGAIN's tools and methodology.

In fact a development environment for CAD can be defined at different levels. You can satisfy the expert programmer with a high level language (for example an object oriented language such as C++) and a certain number of modules (for example classes). On the contrary, you can try to implement specific languages, giving the end-user the possibility to develop its own CAD system, without using a computer science language (for instance, by using visual tools). Many developers are working on new generations of CAD systems intending to become basis to implement application oriented CAD systems. It is in general based on an Object Oriented Methodology, giving a programmer classes and some tools to enrich them. A new generation of commercial systems (CAS.CADE from Matra Datavision, Pelorus from Computervision ...) permit programmers to adapt a CAD basis to their own project.

Our objective is to go further, giving the end-user (and no longer the programmer) tools to adapt a CAD system to its own application. The advantage would be very important, avoiding problems of communication between the end-user and the application programmer. We aim to implement coherent levels of help, facilitating a constructive approach.

In this paper, we emphasise only on the general purposes and on the first step of our implementation (REGAIN V0). This first version is dedicated to the programmer, even if we have experienced some means (graphical formalism in particular) to give a end-user means to develop its own man-machine interface and to architecture the software.

2. MODELLING

2.1. Evolution

CAD models are more and more application oriented, and, less and less geometrical. We are applying this idea in mechanical CAD. When considering, for example, a part as a rough volume and holes, ..., rather than boolean operators on geometrical primitives, the generic model is not CSG neither B-rep, but a more complex structure. Geometry (functional surfaces ...) stays an important support of design and manufacture. We model "conventional" features (holes, bosses ...) that we call "application" features, and consider geometrical entities (cylinder, face ...) as "geometrical" features.

But features modelling doesn't contain, in its classical meaning, a high level semantic. Our goal is to go back to a higher level, considering functional design. For example, a hole has a function (assembly) and the end-user should only have to explain its assembly intention. We are far from that now, but our approach is to construct tools in order to facilitate the integration of this domain oriented methodology.

It seems obvious that a model has two goals :

- to represent the information describing an object, from a given point of view ;
- to provide tools to manage (create, modify, delete, exploit) this information. They provide a service to clients.

The architecture based on generic/applications models can be seen at different levels of details. In fact an application model (at a given point of view) can acquire genericity and serve as generic model for a lower level application. For example, if you have at your disposal a high level of semantic in the generic model, a geometric modeller, such as a B-rep one, is an application modeller. But, when displaying this geometric model, the scene, associated with a basic graphic language, is an application model of the geometric one.

Moreover, the generic information can be distributed on several models. If you need functional surfaces, they will probably be described in a B-rep modeller and the generic modeller will only know the "name" of the functional surfaces.

Anyway, CAD models become more and more complex in themselves. It implies to describe carefully the structure and, in particular in an Object Oriented Approach, the intrinsic and the extrinsic attributes and constraints. When considered as a distributed structure, the product description becomes splitted and need procedures to assure coherence.

2.2 An example

Let consider, as an example, the model of a part [2] : it is structured in objects definition and constraints.

a) objects definition : an hybrid CSG/Brep structure (a graph) whose nodes are either operators or objects (even if operators are considered as objects in an object oriented approach) :

- the operators : boolean operators (geometrical features), extrusion operators (geometrical features, referential, GGI), location operators (geometrical features, referential, GGI) ...
- the objects are : 2D geometrical features (lines, curves, contours ...), 3D geometrical features : surfaces or volumes (cubes, parallelepiped ...), referential (to define measures or positions), GGI.

An objects contains : geometrical information, internal constraints (radius of a circle < GGI ...)

b) external constraints : a graph whose nodes are objects or constraints between objects. This is managed by a constraint propagation algorithm.

It can be seen as an extension of the CSG model to many other kind of operators. Then a B-rep model can be considered as an application model. Faces are evaluated only when necessary, for example for visualisation or dialogue. The only faces which are completely defined in the generic model (hybrid CSG/B-Rep) are functional faces, which are necessary for the definition of the part.

We can then derive views of this model in order to use them for interaction, whose most difficult operation is general picking. We proposed to define a name mechanism between generic an dialogue oriented models as a mean of communication. This name doesn't contain any semantic. For example, if in the interaction, restrictions are given on picking only "holes" and "fillets", this is taken in account by :

- the dialogue manages the picking of a segment (very low level of semantic) and asks the generic model for informations on this object ;
- the generic model answers by the features of the possible picked objects ;
- if it is a hole or a fillet then the interaction can continue else it asks the user for another picking.

3. EXTENSIBILITY

We have seen that the idea of a generic model and applications can be reproduced at different levels within the CAD system. It has proven its efficiency in the particular context of CAD systems [3]. It doesn't mean that the models are centralised. But in a context of concurrent engineering and extended firm, the overall structure of the information becomes evanescent and evolves.

So we propose two concepts :

- the extensibility of models : the goal is to give some tools and methodology to adapt pre-defined models ;
- the extensibility of the scope of the models, including process and overall product information.

3.1. Modellers extensibility

You can't hope to describe exhaustively a modeller, even in a well known domain such as geometry : a given geometric model should be adaptable to new kind of information (for example, a new mathematical surfaces basis) or to specific extensions (for example, add the information of adjacency between faces).

This kind of extensibility is easier to take into account with Object Oriented Languages than with procedural ones. Inheritance and polymorphism give some useful means to do that. But, in order to really implement it, a specific language has to be developed.

3.2. Scope of the models

Considering a domain (for instance CAD), the generic/applications modellers approach can be implemented. When trying to implement concurrent engineering in a networking context, it is necessary to well define the notion of "name". This notion is existing in generic/application modellers, being the basis of communication between models. We have seen in 2. that the scene and the geometric model communicate with a name. As in this case, we propose to manage a structure managing names, containing no semantic at all. In the contrary, a certain number of Product Data Management Systems manage names with or without semantic, as the case may be. It seems to us that a more coherent approach is to consider the naming mechanism as independent and in no case tightly coupled to a certain kind of modeller.

Implementing an independent naming mechanism permits, if wished, to described internal firm rules. Giving a name can be :

- automatic : it will be the case in most intermediary models, i.e. applications models, created by the system to process a certain kind of action such as visualisation, dialogue, calculus ;
- free : the end-user can choice the name;
- firm dedicated : rules are determined for the firm and the user has no freedom or just a certain part of freedom. This last case is very interesting in its ability to prepare a standard way of naming, avoiding redundancy and facilitating retrieving of information. In fact the only semantic that can exist within a name is its signification for a project ...

4. NOTION OF MODELLING CHOICE

Within many systems, the projection of a model on another one (generic/application) is statically pre-defined. As a matter of fact, a B-rep modeller is in general used to represent information to be displayed, for instance, for dialogue purposes.

Our proposal is to let the system decide and, if possible, not to use intermediary non useful models. Let us consider the simplest example : the processing of visualisation from a features-based modeller (fig 1.).

We can distinguish to illustrate our approach three ways (there obviously exist other ways ...) :
(1) features-based modeller to a B-rep and then (for example, with hidden lines processing) to a 2D scene.
(2) Directly features-based modeller to a 2D scene
(3) directly B-rep to pixels (for example ray tracing).

In case (1) interactions between the end-user and the model can be implemented without troubles. A good naming mechanism will give sufficient data to retrieve the corresponding information within the B-rep, and then within the features based modeller. Some manipulations can be associated at the three different levels : at level 1 (scene), drag and drop ... ; at level 2 (B-rep), local operators, volume calculus (approximation) ; at level 3 (features), semantic based operations.

But the approximation imposed by the B-rep can lead to some difficulties (impossibility to correctly model some objects, errors ...). Then, we propose a direct traversal of the features-based model, leading to case (2) and (3).

In case (2), it is possible to implement interactions between the end-user and the features-based modeller using a naming mechanism.

In case (3), no interaction is possible if no specific function is developed within the features-based modeller. This approach is difficult because it needs some other information (camera ...). This case should probably be reserved for high-performance visualisation algorithms, without interactions (else than changing the camera).

We want to attribute more knowledge within models (in this case within the features modeller), explaining how an object should behaves itself under certain circumstances. We have experienced this kind of thing. As an example, we are able to describe behaviours in the model, in order to automatically deduce an adapted man-machine interface. This interface can then be adapted by a end-user, with a graphical formalism [4].

Moreover, we want to describe some rules giving the systems means to decide of its behaviour from the multi-modelling point of view. It should be able to decide, for example, between cases (1), (2) and (3) [5].

In fact, the conversion must also be more application oriented : the ideal would be that giving the constraints of the view, the modeller chooses the adapted conversion using a direct evaluation (direct traversal). For example, if the end-user wants to examine the curvature of a surface or the function of an arm of a robot, it is useless, and perhaps, prejudicial (because of the loss of mathematical properties) to transform the mathematical or the functional model into a B-rep one.

5. TOOL-KITS OF *REGAIN V.0.*

The first version of REGAIN will contain only some tools, but it has to be coherent (we should not have to change this basic tools for future versions). The tools developed in V0 are only programmer oriented. Anyway, they are the basis of the future versions.

We distinguish three tool-kits in REGAIN V.0. for modelling, visualisation and man-machine interface.

Even if we shall try to avoid its use as much as possible (see 4.), we base modelling on an extensible B-rep kernel. Basically, a B-rep class offers services :
- constructors : basic ones (faces, contours, edges, vertices ...) and higher level ones (revolution, extrusion ...) ; the basic ones do not assure validity, in the contrary of the higher level ones.

- operators : boolean operators, using an original algorithm ; our goal is to have at our disposal a general algorithm (manifold and non-manifold objects), ensuring its extensibility (to new kind of objects, surfaces ...).
- primitives to exploit it (services).

Tools to extend the kind of objects will also be provided. For that we provide a methodology and two levels of services :
- level application : it is dedicated to the application programmer.
- level extension : it is dedicated to an expert programmer. Some danger can occur from the use of this service, even if most verification are ensured by the service.

A feature based modeller offer also services in order to define holes ... Two main approaches use features : features recognition and features based design and manufacturing. Having studied in detail these two approaches, we are now developing a feature based model, emphasising on three points :
- use of features in design : we define "design features" and propose a features based model ;
- mapping from design to manufacturing : we propose to consider in general a generic model (features based) and applications models (finite elements, manufacturing ...) and general tools to infer mapping ;
- adding new features into the features library : this point is often badly solved. In fact we try to give an end-user oriented tool in order to construct ("from nothing") its own library. This library extension is from our point of a view essential to obtain operational features based systems.

A viewer is associated with the B-rep model. It provides the user with display tools. It can transform the "generic" model (in this case the B-rep modeller) directly in an image (no interactive operation possible) or in a man-machine interface scene. So, this tool kit can be used in a programming mode or in an interactive mode. The viewer is a good example of basis of algorithms : a careful implementation in object oriented language permits easy and coherent extensibility (i.e. adding a new algorithm is made with the same methodology).

The B-rep modeller and the viewer may seem tightly associated. It is important to notice that the B-rep modeller will become an application modeller in following versions. So only a part of the viewer will be linked to a specific modeller.

Man-machine interface is taken into account by a graphical interactive software. It manages a 2D scene, made of figures. A figure can be based on a display list of lines or contours. Multi-windowing and multi-view porting is available. Its process is based on an event-handling method and persistent or not interactions can be defined. That means that the description of a dialogue is on a relatively high level, describing (possibly in a library) applications oriented interactions (for instance : on a basic level, an interaction to recover a point, on a high level, an interaction to describe a complete specific part).

An object must contain its methods of construction, visualisation, ... In fact, it must contain at least one method for each valid function. So you can have two classes of dialogue, which can be deducted automatically from the models :
- the object dialogue : when pointing on an object, the object itself is able to display the possible actions as soon as it is identified. This display can be either menus or a behaviour of the object (changing colour, flicking, symbol ...) ;
- the actions dialogue : when entering an action some different processes can be proposed to the end-user. The dialogue can take into account the process of the action and some specific constraints (for example, it is of the responsibility of the man-machine interface to assure that the end-user gives a positive value to the thickness of a solid).

Certain parts of the dialogue will then be automatically deducted from the models but others have to be defined by the end-user utilising a graphical formalism.

Rules of development are part of this version. they concern the implementation of an application or the extension of a tool-kit. The chosen language is C++. This first version is primarily programmed on PC, with Windows NT. In order to facilitate interoperability, it will support OLE Automation.

6. CONCLUSION

CAD systems development seem to be changing. They have to take into account complex applications within a broader context :
- global firm constraints (productivity, time to market, cost, quality ...),
- interoperability ;
- extensibility and adaptability ;
- process modelling.

They have also to solve technological problems, concerning their architecture (should be extensible, adaptable ...) and their semantic.

We think that the actual development are semantically too poor (the classical notion of feature ...).

Three important evolutions will occur in the next few years :
- CAD systems will no longer be monolithic, but will become basis of development. A first family will be based on programming tool-kits ; some years later, end-user dedicated development environment will be available. A lot of work has to be done on man-machine interfaces.
- the integration of simulations will impose a higher level of semantic. We will no longer speak in terms of features but in terms of domain solutions [6].
- interoperability is an obliged step : CAD systems are no longer isolated, but they are part of the decision process in the enterprise.

To achieve these goals, multi-modelling is essential. The problem is that it is difficult to define exactly what is multi-modelling. It is easy to understand that independent tool-kits, providing a well defined service and a methodology of implementation are a basis of multi-modelling. But the notions of mapping (generic/applications) are evolving when considering a larger context (interoperability, networking).

7. REFERENCES

1 Y. Gardan, A new approach to develop adaptable CAD systems, EDA 97, Banglok, march 1997

2 Y. Gardan, "A multi-modeller approach for man-machine interface and graphics in CAD" Proceedings International conference Pacific graphics'94/CADDM '94 - August 26-29, 1994, Beijing, Chine

3 Y. Gardan, "Defining CAD Systems a adaptable to design process -"ETFA'95 INRIA/IEEE Proceedings Conference on Emerging Technologies and Factory Automation - 10-13 octobre 95 - Paris

4 Y. Gardan, JP Jung, B. Martin, "an end-user approach to design man-machine interface for CAD/CAM", in proceedings of IEEE International Conference on Systems, Man and Cybernetics, Le Touquet, France, octobre 93, 525,530

5 Y. Gardan, B. Martin, I. Stemart, "Behaviors : from man-machine interface to design for manufacturing" - CESA 96 - 8-12 juillet 96 - Lille - France

6 P. Béziat, Y. Gardan, "Technical Solutions for Mechanical Design and Manufacturing" - 28th International Symposium on automative Technology and Automation Stuttgart (ISATA) - 18-22 sept 95- Stuttgart Germany

ON DEVELOPING A FLEXIBLE FIXTURE FOR PLANAR OBJECTS

G. C. I. Lin and H. Du
Centre for Advanced Manufacturing Research
School of Engineering
University of South Australia
The Levels, SA 5095 Australia

ABSTRACT

This paper presents the development of a three fingered automated flexible fixturing system for planar objects in the machining process. The three fingered fixturing system consists of two computer numerically controlled (CNC) modules and a fixturing algorithm. One of the modules has two fingers positioned along two circumferences of two adjacent circles, while the other module has one finger which can be adjusted along a slot. With the fixturing algorithm, flexibility in the fixturing system is achieved by reconfiguring the fingers of the system to accommodate workpieces of different shapes and sizes. A prototype of the system has been built and tested. Examples are given of the application of the system to workpieces of different shapes. The system is shown to be flexible, reconfigurable and fully automatic, capable of fixturing planar objects of different shapes and sizes in the machining process.

KEYWORDS

Fixture, Flexible Fixture, Fixturing System, Fixture Design, Workholding technology

1. INTRODUCTION

Fixturing is a fundamental characteristic of most production operations, such as machining, assembly and inspection. Once the shape of a part and its desired position and orientation are known, fixtures are usually custom designed by manufacturing engineers and machinists in the industry. With the advance of modern technologies and the competitive world market, these custom designed fixtures are not only time-consuming and costly to build, but they do not have the flexibility to deal with parts or assemblies of different shapes and sizes. This is frequently the cause of bottlenecks in the modern manufacturing industry. Flexible fixtures, which are adaptable to accommodate a variety of parts and assemblies of different shapes and sizes, have been suggested as an obvious solution to overcome this problem.

Different kinds of flexible fixtures have been proposed or developed in the last twenty years. An overview of the methodologies of fixturing has been presented by Thompson and Gandi [1]. A large proportion of this research actually concentrated on the development of the automation of fixture design using modular fixture kits or modular elements [2-5]. A modular fixture kit consists of many elements, such as baseplate, locators and clamps. A custom designed fixture can be assembled by using the components from the kit. Even though such a modular fixture has been widely applied in industry, it can only fixture a limited number of parts and has to be set up either manually or by robots [5]. To overcome these limitations, a new approach may be adopted.

Research has been conducted into the minimum number of fingers needed to immobilise an object in robot hand grasping; this has provided a new approach to flexible fixturing. Markenscoff and others [6] have established that four contact points suffice to immobilise a generic 2D object, and seven suffice to immobilise a generic 3D object. On the other hand, Czyzowicz and others [7] have shown that generic 2D

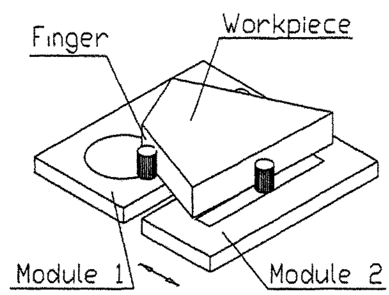


Figure 1 The three fingered fixturing system

and 3D polygonal objects can be immobilised respectively by three and four frictionless point contacts. The use of the minimum number of fingers in a fixture has two advantages. One advantage is that it may lead to more adaptable or more efficient fixturing techniques and fixturing planning algorithms. The other advantage is that the possibility of using fewer fingers may lead to more efficient re-fixturing methods in a machining process involving frequent repositioning of fixtures. Hence, it is desirable to implement the concept of the minimum number of fingers in fixturing.

Based on the minimum number of contact points for a 2D object, a three fingered flexible fixturing system is proposed by the authors and is currently being developed as shown in Figure 1. The three fingered flexible fixturing system consists of two computer numerically controlled (CNC) modules, one fixed and the other movable. The fixed module has two fingers positioned along two circumferences of two adjacent circles, whereas the movable module has one finger which can be adjusted along a slot.

With the fixturing algorithm in this development, the flexibility of the fixturing system is achieved by reconfiguring the three fingers of the system. Therefore, the aim of this research is to develop a CNC three fingered fixturing system that can configure itself automatically and has flexibility to deal with planar objects of different shapes and sizes. One contribution of this research is that the idea of the minimum number of fingers to immobilise an object is used in developing the two fixturing modules. Another contribution is that to integrate with other systems in an FMS, an approach has been taken which integrates with a CAD system to develop a fixturing algorithm for polygonal objects. Even though the three fingered fixturing system deals with planar objects in this paper, the approach used can be extended to 3D workpieces.

This paper presents the detailed mechanical design and a fixturing algorithm for the three fingered flexible fixturing system, along with examples of applications of the system. Finally, conclusions are given as to the effectiveness of this design, and future developments anticipated for the system are described.

2. THREE FINGERED FLEXIBLE FIXTURING SYSTEM

2.1 Overview of the System

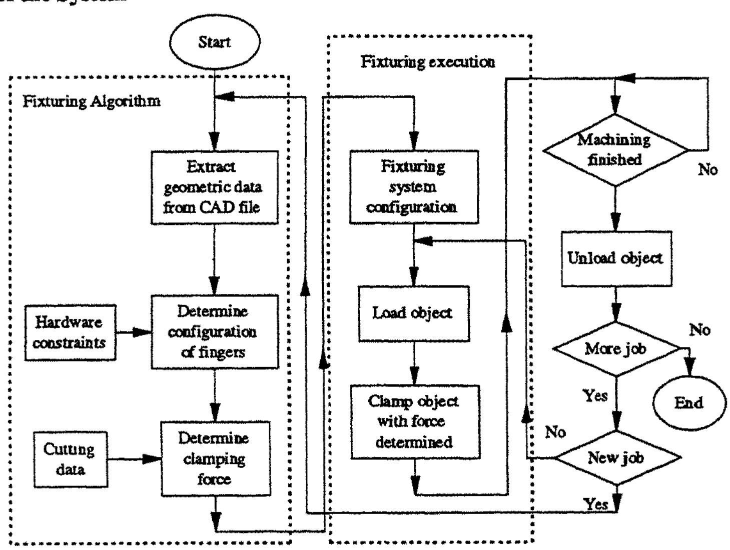


Figure 2 The system diagram of the fixturing operation

The whole system proposed consists of the fixturing hardware and algorithms for fixturing planar objects. Figure 2 shows a system diagram of the whole fixturing operation. Once the CAD data for a workpiece is known, the fixturing algorithm is used to determine the positions of the three clamping points on the workpiece. According to the three clamping points, configurations of the three fingers are determined based on the constraints of the three fingered system. The clamping force is determined from the positions of the three clamping points and the cutting data. At the fixturing execution stage, the system configures itself automatically according to the configurations of the fingers. Once the workpiece is loaded, the fixturing system clamps the workpiece with the amount of the clamping force determined previously. This fixturing process can be controlled directly from a machining centre.

2.2 Mechanical Design of the Three Fingered Fixturing System

2.2.1 Conceptual Design of the System

As shown in Figure 3, the system consists of two modules. The fixed module is called Module 1 and the other movable module is called Module 2. To achieve greater flexibility with the three fingers, positions of the fingers need to have large range of adjustments. Figure 3 shows that fingers 1 and 2 can be positioned along the circumferences of two circles (C_1 , C_2) respectively, having adjustments of $\pm r$ in both x and y axes, whereas finger 3 can be adjusted in only y axis direction with the adjustment of $\pm h$. Moreover, Module 2 can be positioned along x axis. The positions of the fingers and Module 2 are completely depended on the shapes and sizes of objects to be clamped.

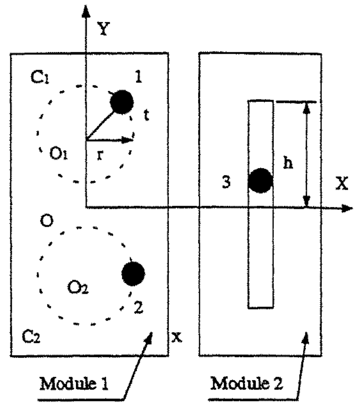


Figure 3 Fixturing adjustments

2.2.2 The Fixed Module

To implement the above conceptual design, the same approach as used by the authors [8] for the previously proposed flexible fixture for 3D objects, is taken to develop the fixture. A detailed design of Module 1 is made under consideration of strength, size, weight and cost. Figure 4 shows the detailed design of Module 1.

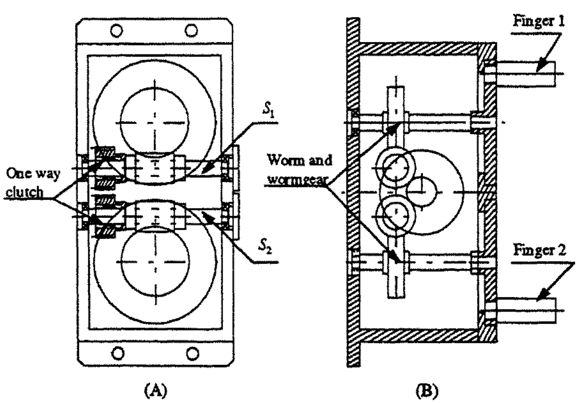


Figure 4 Detailed design of Module 1
(A) Front view, (B) Left side view

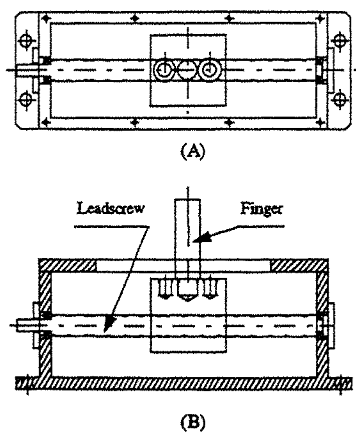


Figure 5 Detailed design of Module 2
(A) Top view, (B) Front view

In this design, the mechanisms of the two fingers are designed to be the same except that the two one-way clutches are set in opposite directions on the two shafts (S_1, S_2). Therefore, the positions of the two fingers can be controlled by a step motor to rotate in opposite directions. The two sets of worms and worm gears are chosen in such a way that the positions of two fingers will be self-locked once configurations are finished. The two fingers are designed to withstand the maximum clamping force of $2000N$ during the machining process.

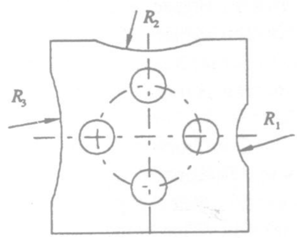


Figure 6 Specially designed finger

2.2.3 The Movable Module

Figure 5 shows a detailed design of the movable module. A leadscrew is used to transfer the motion from rotation to translation. One step motor is used to control the position of the finger. The nut of the leadscrew is designed in such a way that a variety of fingers, such as a specially designed finger (discussed below), can be installed to achieve greater flexibility.

2.2.4 One Specially Designed Finger

A finger has been specially designed and developed to explore the idea of surface contacts between the finger and an object in fixturing. Figure 6 shows a 2D projection of the finger. It has three different concave surfaces and one flat surface on its edges. The radii of the concave surfaces (R_1, R_2, R_3) are chosen based on the research conducted by Rimon and Burdick [9]. By using different contact surfaces of fingers, greater flexibility can be achieved to immobilise objects. Fingers with convex surfaces will be developed in the future.

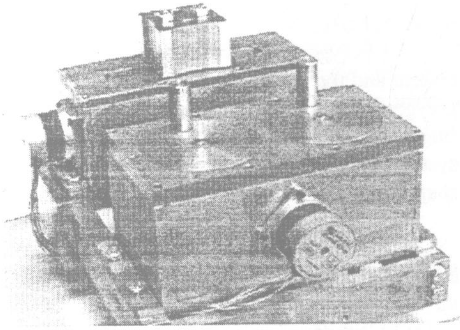


Figure 7 Prototype of the three fingered fixturing system

2.2.5 Prototype

A prototype of the system (Figure 7) has been built and tested to clamp various shapes of objects. The overall mechanical design of the system is satisfactory. The one-way clutches work as expected so that one step motor can drive the two fingers to their desired positions in Module 1. The fingers show good rigidity. The specially designed finger shows that it can prevent a workpiece from loading in wrong position in the fixturing, as the workpiece will be pushed back to the position during the loading process. However, due to the backlashes in the gears and the rigidity of the worms and wormgears in the fixed module, the positioning accuracy of fingers is not high when forces are acting on them.

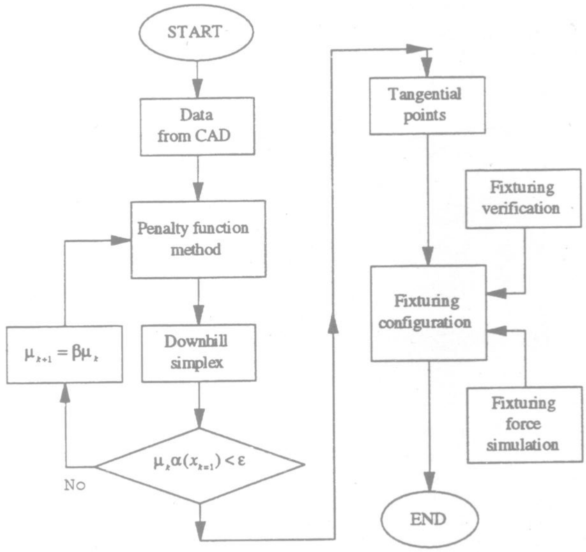


Figure 8 Algorithm diagram

2.3 The Algorithm

Figure 8 shows a diagram of an algorithm. The principal idea of this algorithm is the implementation of the concept of the maximal inscribed circle of a polygon. As shown in Figure 8, once the geometry of the data of a polygon is extracted from a CAD file, the algorithm applies the penalty function method [10] and the downhill simplex method [11] to determine the maximal inscribed circle in the polygon. The Newton-Raphson method is then used to obtain three tangential points where the maximal inscribed circle meets the sides of the polygon.

Configuration of the three fingers in the system can then be determined from these three tangential points. Optionally, the fixturing configuration can be verified and simulated by using two software modules developed previously by Du and Lin [12]. The algorithm has been implemented in a program written in C language for Windows. More detailed discussion about the algorithm can be found from Ref. [13]. However, due to the constraint of this algorithm, it can only deal with convex polygons. A general algorithm which can deal with other types of polygons will be developed in the future.

3. EXAMPLES OF APPLICATIONS OF THE SYSTEM

A polygonal object and a disk are used as examples to show the flexibility of the three fingered system and the algorithm developed by the authors [13]. Even though the algorithm can only deal with polygonal objects at this stage, the fact that the three fingered fixture can clamp a disk indicates that the system is capable of fixturing objects of different shapes.

The CAD data for a planar object with four edges is generated by a CAD package as an input for the algorithm developed. The coordinates of the vertices of the object are (0,0), (191, 0), (191, 55), and (0, 246). Figure 9 shows one of the fixturing configurations of the three fingers and the results of the calculation. The results include the coordinates of the vertices of the object and the three fingers, and the distance between the two modules.

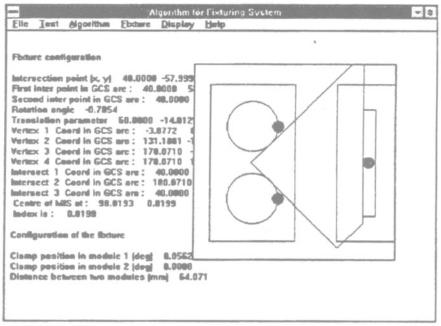


Figure 9 Results of fixturing configuration

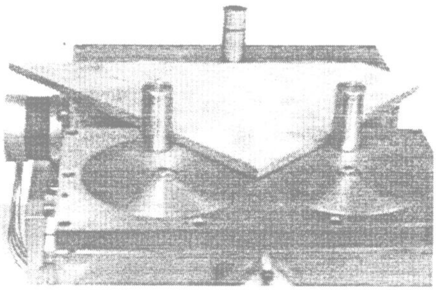


Figure 10 Fixturing example

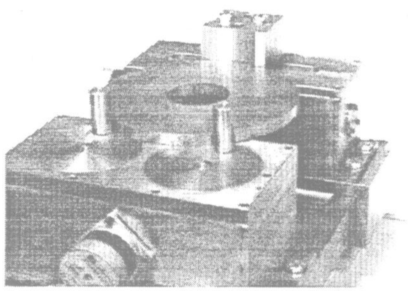


Figure 11 Fixturing example

The results of the fixturing configuration are used to configure the three fingers. Once the object is loaded, Module 2 clamps the object firmly with the amount of clamping force determined by the cutting force model in the algorithm. Figure 10 shows that the object is clamped by three fingers. Figure 11 shows another example in which a disk with a diameter of $200mm$ and height of $12mm$ is fixtured by the system. It shows that the system is able to fixture workpieces of different shapes and sizes for the machining process.

4. CONCLUSIONS AND FUTURE DEVELOPMENTS

In conclusion, a new approach towards flexible fixturing has been taken to further explore the potential of its flexibility and automation in manufacturing. As a result, a three fingered flexible fixturing system has been developed for planar objects in the machining process.

The three fingered system includes two CNC modules and a fixturing algorithm. By configuring the fingers in these modules according to the fixturing algorithm, objects of different shapes and sizes can be automatically clamped by the fixture. An auxiliary mechanism clamps the object with the amount of clamping force determined by the cutting force model. The mechanical design of the fixture and the algorithm are presented in this paper. A prototype of the system has been built and tested, showing satisfactory performance in mechanical design. Two examples of applications of the fixturing system demonstrate its flexibility. The system has shown to be flexible, reconfigurable and fully automatic. It is capable of clamping planar objects of different shapes and sizes in the machining process.

Further development will involve implementation of transducers in the hardware system to control the clamping force. The use of strain gauges in the fixturing system will also monitor the cutting force and clamping force during a machining operation. The use of sensors will provide a more intelligent fixturing system. Further work will also include integration between the three fingered fixturing system and a machining centre.

5. REFERENCES

1. Thompson, B. S. and Gandhi, M.V., "Commentary on flexible fixturing", Applied Mechanical Review, 39(9), pp. 1365-1369, 1986.
2. Chan, K.C., Benhabib, B., and Dai, M. O., "A reconfigurable fixturing system for robotic assembly", *Journal of manufacturing systems*, 9(3), pp. 206-221, 1990.
3. Gandhi, M.V., and Thompson, B.S., "Automated design of modular fixturing for flexible manufacturing systems", *Journal of Manufacturing Systems*, 5(4), pp. 243-252, 1986.
4. Liu, C-L., "A Systematic Conceptual Design of Modular Fixtures", *International Journal of Advanced Manufacturing Technology*, 9(4), pp. 217-224, 1994.
5. Youcef-Toumi, K. and Buitrago, J.H., "Design and implementation of robot-operated adaptable and modular fixtures", *Robotics and computer-integrated Manufacturing*, 5(4), pp.343-356, 1989.
6. Markenscoff, X., Ni, L. and Papadimitriou, C.H., "The Geometry of Grasping", *International Journal of Robotics Research*, 9(1), pp.61-74, 1990.
7. Czyzowicz, J., Stojmenovic, I. and Urrutia, J., "Immobilizing a Polytope", Lecture notes in Computer Science, Vol. 519, pp.214-227, 1991.
8. Lin, G.C.I. and Du, H., "Design and Development of an Automated Flexible Fixture", Proceedings, the 4th Internal Conference on Automation Technology (AUTOMATION '96), Hsinchu, Taiwan, pp.475-480, 7-10 July, 1996.
9. Rimon, E. and Burdick, J., "New Bounds on the Number of Frictionless Fingers Required to Immobilise Planar Objects", *Journal of Robotic Systems*, 12(6), pp.433-451, 1995.
10. Bazaraa, M.S. and Shetty, C. M., Nonlinear Programming: Theory and Algorithms, pp.336-342, John Wiley & Sons, 1979.
11. Press, W., Flannery, B.,Teukolsky, S. and Vetterling, W., Numerical Recipes in C, pp.305-309, Cambridge University Press, 1979.
12. Du, H. and Lin, G.C.I., "Clamp Location Optimisation in Fixturing Configuration", Proceedings of the 12th International Conference on CAD/CAM Robotics and Factories of the Future, Middlesex, UK, pp.1054-1059,1996.
13. Lin, G.C.I. and Du, H., "Fixturing Planar Objects By Using the Maximal Inscribed Circle in a Polygon", Australasian Conference on Technology for Manufacturing, 1997, to appear.

AUTOMATED CORRECTION OF SHAPE ERRORS IN MANUFACTURING PROCESSES

Bert van den Berg,
IMTI, National Research Council of Canada,
Building M3, Montreal Road, Ottawa Canada

ABSTRACT

Processes for manufacturing complex shapes often require 'fine-tuning' after they emerge from CNC machine tools. This adjustment or benching, is typically done by skilled technicians. Computer Integrated Manufacturing allows for the elimination of some of this handwork, improving the creation of the tools, and adding to the understanding of today's manufacturing processes. This new (quantified) understanding may lead to better analytic methods for accurately predicting the shape of parts produced by a given set of manufacturing processes.

This paper presents a method of applying computer-integrated shape error correction to tooling for formed tooling used in Electro-Chemical Machining or ECM. In this method (called TOPAC-ECM), automated modifications to the CNC toolpath are made based on measurements of a first-off part produced by the electrode. Besides electro-chemical machining, this method can be applied to a variety of surface contouring processes. This approach offers increased precision, and reduced product development time and cost. The paper reviews other computer integrated methods for shape error correction, and then describes the method developed.

KEYWORDS

Shape Error Correction, Adaptive Control, Computer Integrated Manufacturing, Closed-loop Machining, Rapid Prototyping

1. INTRODUCTION

An important step in the manufacture of discrete parts is the development of the process parameters to acceptably produce a particular type of part (sometimes called setup, or qualification). In this important step, the production process parameters that were established using *best-practice estimates*, are adjusted based on observations made from the process, and the components it produces.

The performance of a process can be characterized by its repeatable and random components. Thus, a machining process may have a significant and repeatable error due to such features as tool run-out, thermal growth, servo-error, and axis misalignment. Random errors might arise from vibration of the machine due to nearby moving vehicles, variations in the material to be machined, and facility temperature variations. Some rapid prototyping systems have repeatable errors due to the volume changes as the material solidifies, and the kinematic errors exhibited by traditional machine tools. Forming processes such as casting, moulding, stamping, have repeatable errors due to die shape, and residual stresses developed in the formed part. Other processes such as electro-discharge machining, and electro-chemical machining have repeatable errors due to tool shape, hydrodynamic, and electrical potential affects on the machining zone.

Computer Integrated Manufacturing offers a method to improve process adjustment. The general approach is shown in figure 1.The literature presents a variety of applications of shape error correction.

2. LITERATURE REVIEW

Early work includes efforts at extracting rigid-body best fits of parts which has been applied to adapting process parameters (toolpaths) to suit the actual location of the part [see Bourdet'76, van den Berg'87]. Continued challenges in defining the "best-fit" are evidenced by Chatelain's work to develop a fitting method to provide optimal machining stock [Chatelain'95].

Duffie et al's work focus on tool and die grinding and polishing [Duffie'94] although it could be applied more broadly. In tool grinding, the grinder must remove the machining cusps to reveal the finished tool shape. The approach taken was to measure the actual part, and then calculate a deforming least-squares fit of the design surface to the measured points. From the deformed surface a toolpath was generated that better matched the part shape. Duffie et al extended this work to a concept of tri-cubic solids [Duffie'88].

Metal drawing for applications such as autobody panels has been the subject of substantial research. These include work by Cripps et al to model the springback of metal stampings based on empirical measurements [cripps'91]. They attempted to relate the compensation applied (overcrowning) to the springback observed. Hardt et al have developed methods to compensate springback in sheet forming by directly changing the tool shape using a novel tooling method [Webb'91].

Airfoil shapes such as tooling attract significant research on improving form accuracy, possibly due to their high value. Work has been done for reprofiling of airfoil tips [Fleisig'96, Rigid'93] to match the neighbouring *original* material (based on measurements are taken near the repair zone). Toolpaths are created from surfaces which blend the design intent and the measurements on the actual part. Note that service-exposed airfoils vary in form and position from their design specifications.

Recently Chasse and van den Berg have developed a new method for correcting shape errors based on finite element (FE) analysis [van den Berg'95]. This method models differences between a manufactured part and the desired shape using a physical analogue such as force on a FE model. The inverse of this analogue is applied to the FE model to predict the shape required to create an accurate part.

Work by Gerchman et al have applied an idea similar to that described herein for correcting the form errors on aspheric optics [Gerchman'91]. They machined the part to near the desired shape (pre-finish), measured the resulting part, and compensated the final toolpath based on the errors observed in the pre-finished part.

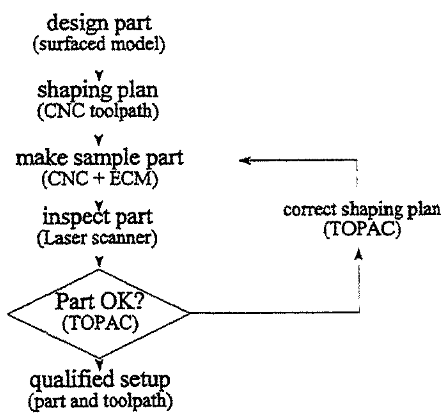


Figure 1: Process Setup Sequence

All the methods described above improve our understanding of the corrections made to manufacturing processes to produce useful parts. There is a complement body of active research that attempts to develop analytic models of manufacturing processes, so that the resulting part shape and performance can be predicted and corrected before the actual process has been applied. At present, it would appear that analytic approaches provide useful insight into defect creation (e.g., voids, sinks, tears, surface roughness, cracks), but lack the ability to model the fine, but important shape errors that must be corrected to produce useful parts.

3. TOOL PATH CORRECTION

3.1 Computer Defined Shaping

Generally there are two computer-controlled ways of defining part shapes: Computer Numerical Control (CNC) machining, and rapid prototyping (RP)[1]. The attributes of the two processes define the possibilities for computer-integrated shape error correction. Thus for CNC machining the tool trajectory, feeds, speeds are defined in the CNC program and can easily be adapted. In current RP systems, the nodes and facets of the part model, or the contour and filling patterns of the slice layers are defined in the RP construction file(s). The approach described here uses CNC machining to produce ECM electrode shapes with a ball nose cutter. Thus, we must affect the milling process to alter the shape produced by the ECM process, particularly the toolpath trajectory. The TOPAC method changes the position of each cut point in the toolpath to improve the part's shape accuracy.

3.2 Part Shaping Using Electro-Chemical Machining

ECM uses a large current passing between an electrode, and a (metal) blank. The current is carried between the two elements by a flowing aqueous salt solution. This generally creates good conditions for ionic dissolution at the blank surface, particularly where the electrode shape is nearest to the blank. As the electrode is driven toward the blank, the blank material dissolves away, assuming a shape that is nearly the obverse of the electrode. Parameters that affect the shape assumed by the part include fluid flow rate, fluid pressure, aqueous ionic concentrations, presence of precipitates, voltage, current, electrode velocity and fluid temperature [Jain'91,Chetty'87, Ruzjac'91]. These parameters must be established and maintained for repeatable ECM operation. The local shape of the electrode affects the local electrical potential across the electrode-part gap and plays an important role in part shape. Therefore, the shape of the part can be affected by changing the trajectory of the toolpath used to cut the electrode

The starting shape is the negative of the shape to be produced. However physical and chemical features of the ECM process mean that subtle changes are required to the ECM electrode. As a first step, an electrode with the negative of the required part shape is created. The part is measured to identify deviations from the desired part's shape. In the manual approach, a toolmaker assesses the error information given, and then modifies the electrode shape using tools such as sandpaper, grinders and files. A new part is then produced to test the changes. This cycle is repeated until satisfactory parts are produced. For parts such as airfoils where shape tolerances may be +/- 50 micro-meters or smaller, even skilled operators have difficultly making the required (fine) adjustments to the electrode.

In the computer integrated method, the measured points from a sample part are compared to the desired shape, and the differences used to modify the CNC toolpath. The actual method will be described below in more detail.

3.3 Computer Integrated Toolpath Correction Algorithm

---

[1] The work by Hardt et al is an example of a computer controlled process where the tool shape is directly formed from pins, rather than using CNC or RP to fabricate a die. However, this type of approach appears to be limited to the laboratory.

The general toolpath correction algorithm is presented below. It assumes that:

- the sample part's differences in shape from the desired part's shape are small with respect to the local/regional curvature of the part's surface.
- the sample part's surface was created using an electrode machined from an (available) CNC toolpath without significant human modification of the surface,
- the cutting tool contacts the desired part's CAD surface in a single location at each cut point.
- a set of point rows has been measured on the sample part's surface,
- the desired shape's CAD model is available, and
- the manufacturing process(es) to be corrected is/are repeatable.

0) For all cut points in the (linear motion) toolpath (i=1..n):

- calculate a minimum distance point (MDP_i), and surface parameters (U_i,V_i) from the tool centre point (TCP_i) to the desired shape's CAD surface using an Iterative Closest Point algorithm (ICP). The algorithm uses a faceted model of the surfaces to locate the surface patch of interest, and then an interactive closest point calculation to estimate MDP_i. The ICP algorithm is omitted for the sake of brevity.

1) For all measured points (j=1..m):

- calculate MDP_{jl} from a measured point (MP_{lj}) to the desired shape's surface using an ICP method. Save the surface number and parameters (U_{jl},V_{jl}) corresponding to MDP_{jl}.
- define the vector from the MDP_{jl} to MP_{jl}.
- for measured points given as probe centre points, add to this vector a vector from MP_{jl} towards MDP_{jl} of length equal to the probe radius. The resulting vector is the error vector(γ_{jl})
- evaluate this vector against the local surface normal to define inside/outside, and against the form tolerance to define quality status.

2) If $\forall$ γ_{jl}, $\gamma_{jl} \leq$ the form tolerance, then the part, process, and electrode are satisfactory (stop the process). Else go to step 3.

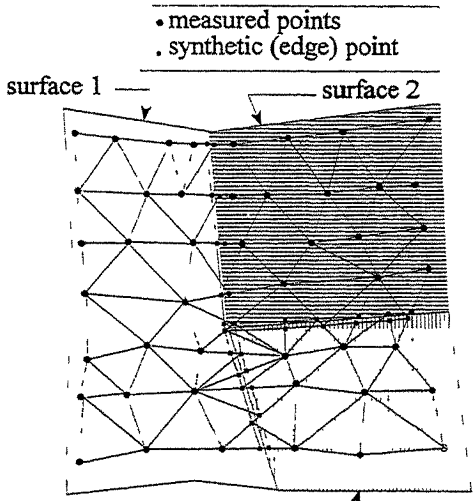


Figure 2: Sample Tesselation

3) From the data on the measured points(1..m):

- tessellate the measured points. The points must be measured in rows (with variable number of points permitted in each row). This allowed the tessellation to be conducted in XYZ space. A sample tessellation is shown in Figure 2. This algorithm is not presented here for the sake of brevity. Alternatively the measured points could be tessellated in UV space on each surface. The tessellation would have to be followed by a further step to provide means of tessellating across surface boundaries.

4) From the tessellated data and the error vectors, build a response function using a quasi-Cartesian representation (U_{jl},V_{jl},γ_{jl}), where the U,V axis are aligned with surface parameters, and γ_{jl} is along the local surface normal.

5) For all cut points in the toolpath(i=1..n):

- identify the surface, and facet in the response map corresponding to a cut point (U_{il},V_{il})
- interpolate the nominal correction ($|C_{il}|$) as the negative of the length of the local error from the

response function (see figure 3).

- if one correction cycle has already been applied (l=1) then the correction size $|C_{il}|$ is extrapolated further based on a linear extrapolation as indicated below:

  $|C'_{il}| = \gamma_{il} + (\gamma_{il} - \gamma_{i,l-1})/|C'_{i,l-1}|$...(eq. 1)
- if two correction cycles have been applied (l=2) then the correction size $|C_{il}|$ is extrapolated based on a quadratic extrapolation as shown in Figure 4.
- finally, if three or more correction cycles have been applied ($l \geq 3$) then the correction size $|C'_{il}|$ is extrapolated based on a cubic extrapolation.

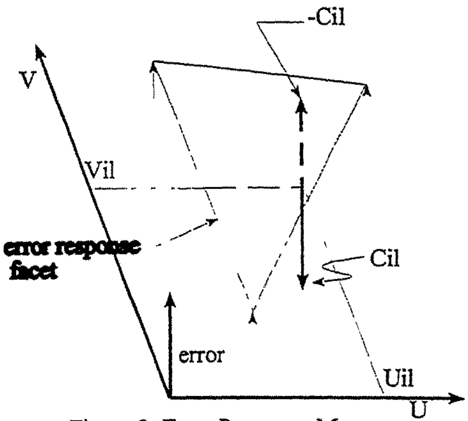


Figure 3: Error Response Map

- define the modified tool centre point:

  $TCP'_{il} = TCP_{il} + C_{il}$ . . . (eq. 2)

  where C_{il} is a vector of size $|C_{il}|$ along the direction of the surface normal at MDP_{il}. Thus, TCP_{il} is moved by a C_{il}. Note that for $l>0$, C_{il} is replaced by C'_{il} in equation (2) above.
- output a new cut point in CNC coordinates based on TCP'_{il} as calculated above.

6) return to step 1) and evaluate the next sample part.

The result of this algorithm is a new toolpath with each cut point's displacement related to the local error on the part measured.

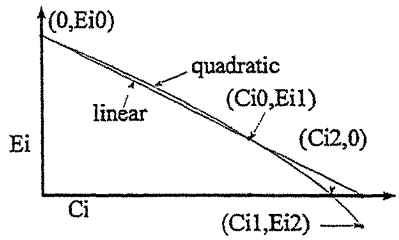


Figure 4: Quadratic Correction

4 RESULTS

The algorithm above was implemented in a program known as TOPAC-ECM (Tool Path Correction for Electro-Chemical Machining). It was tested on two machined parts, and several ECMed shapes (airfoils). The results show that the program successfully corrected for shape error, regardless of whether the errors were due to the machining process or ECM process. In all cases the approach provided satisfactory results after one correction iteration. For machining, the errors measured after one cycle were with the range of non-repeatable errors of the CNC mill and measurement processes. For ECM, significant time savings can be expected since the TOPAC-ECM approach reduces the number of fine-tuning iterations required. Further, the modified CNC program used to machine the final electrode can later be reused to duplicate, or repair a worn or damaged electrode (capturing this knowledge).

One significant limitation with this method is the requirement that the local errors be small with respect to the local curvature. To better understand this, consider the two-dimensional example shown in Figure 5. The problem occurs at the sharp bend (fillet radius). In this region the error vectors have a magnitude approaching the local surface radius of curvature and the form of the fillet has been lost. One might imagine a different strategy being applied in these regions.

5 CONCLUSIONS AND FUTURE WORK

Shape error correction based on local surface measurements is a powerful, automated means of fine-tuning a contoured shape.

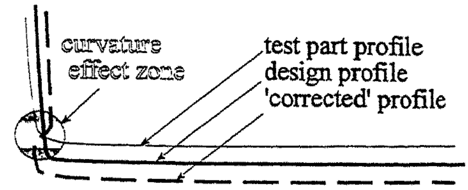


Figure 5: High Curvature Effect

Besides the limitation described in section 4, this work, and other correction methods do not deal well with surfaces for which there is no measurement data. Methods such as Chasse and van den Berg's Finite-element-based corrections, and Duffie's tri-cubic solids point to possible approaches.

6. ACKNOWLEDGEMENTS

This work was partially funded by Walbar Inc.. Some of the software was written and tested by D.A. (Doug) Zhou.

7. REFERENCES

Bourdet P., Clement A., *Controlling a Complex Surface with a 3 axis Measuring Machine*, Annals of CIRP Vol 25/1, pp. 359 - 361, 1976.

van den Berg B., Orban P., *Automated Part Setup for Machining of Sculptured Surfaces*, Proceedings of Maple'87, Winnipeg Manitoba, 1987.

Chatelain J.F., Systeme d'analyse de donnees d'inspection tridimensionnelles integres a un systeme CAO/FAO, PhD Thesis, Ecole Polytechnique, Montreal Quebec, 1995.

Duffie N.A., *Error Compensation Algorithms for Sculptured Surface Production*, Transactions of ASME, V116, pg. 144-152, 1994.

Duffie N.A., Feng S., and Kann J., *Error Analysis and Manufacturing Proces Compensatin using Tri-cubic solid databases*, Annals of the CIRP, v37/1 1988.

Cripps R.J. Barley S.A., *A Geometric Characterization of Springback in Drawn Panels*, Math Engineering Industry, V3, N3, pp 205-214 1991.

Webb R.D., Hardt D.E., *A transfer Function Description of Sheet Metal Forming for Process Control*, Transactions of the ASME V116, pp 44-55, 1991.

Anonymous, Rigid Machine Tool Technical Information Blade Repair: Milling of Welded Aero Engine Blades, 26071 Merit Circle, Suite 112, Laguna Hills, Ca 92653, 1994.

Fleisig R.V., Spence A.D., *CAD/CAM integration for Five-axis closed loop machining*, pp 402-409, Proceedings of CSME Forum in 13 Symposium on Engineering Applications of Mechanics, McMaster University, Hamilton Ontario, 1996.

van den Berg B., and Chasse D., *F.E. Approach in Shape Adaptive Manufacturing*, Proceedings of 15th Canadian Congress of Applied Mechanics - CANCAM 95, Victoria BC, 1995.

Gerchman M.C., *Compensation of Residual Form Errors in precision machined components*, SPIE V1573 Commercial Applications of Precision Manufacturing at the Sub-Micron Level (1991), pp 201-239.

Jain V.K., Rajurkar K.P., *An Integrated Approach for Tool Design in ECM*, Precision Engineering, April 1991, V13, N2, pp 111-124.

Chetty, O. V. K., Rao, G. P., Narayan R.; *Flow Velocity Effect on Electrochemical Anodic Smoothing*, Journal of the Institution of Engineers (India), Mech. Engineering Division 3 v p 49-51,68 Nov. 1987.

Ruszaj, Adam, *Investigations on the Process of Electrochemical Sinking, Taking Into Account the Randomness of Phenomena Occurring in the Machining Area*, Wear v 147 n 1 25-40, Jul. 5 1991.

A STUDY ON AN INTEGRATED CNC SYSTEM FOR TOOL GRINDING

Yanhong Zhou Jian Li Chenpei Wang Ji Zhou
School of Mechanical Science and Engineering
Huazhong University of Science and Technology
WuHan, P.R.China

ABSTRACT

An integrated CNC system for grinding tools and cutters has been developed on the basis of 80486 industrial control computer. The system is incorporated with an automatic programming software to grind a very wide range of either special or standard tools. The grinding movement simulation and verification; the direct interpolation for the complicated tool-path and the dynamic display for the machining process have been implemented on the system software. This paper presents the architecture, characteristics and working principle of the software system. Some key techniques of the grinding movement simulation and the direct interpolation control for the complicated grinding movement path are given.

KEYWORDS

CNC; cutter; grind, CNC program

1. INTRODUCTION

With the widespread application of high-performance CNC machines and the machine center, the demand for grinding and regrinding a wide variety and a large number of precise tools has been increased in the industry enterprises. We have developed an integrated CNC system software for grinding tools and cutters which can be used to control 2-5 axes tool and cutter grinder to machine cylindrical or tapered mills, reamers, hobbing and profile milling cutters, etc. The structure dimensions of cutters and the processing parameters can be directly input by operators on the CNC graphic menu. Except the general and standard cutters, the nonstandard cutters or other components defined by users can be also ground by modifying.the parameters on the screen. The system has the function of simulating the grinding movement and verifying interference and collision between the wheel, the workpiece and the fixture. The complex grinding movement has been accomplished by the real-time interpolation, which can approximates the surface contour with the tiniest NC step. The system software has been equipped with the CNC tool and cutter grinder, and some qualified tools and cutters have been acquired in the workshop successfully.

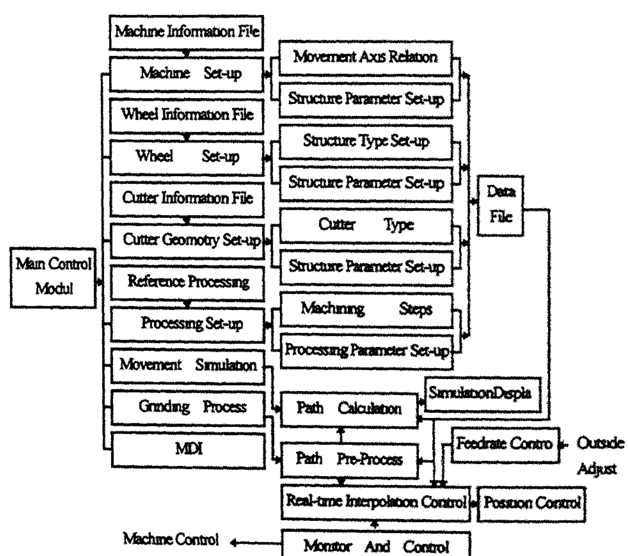


Fig. 1 The system's structure

2. THE SYSTEM' S STRUCTURE AND FUNCTIONS

The system is composed of some module such as the machine set-up, the grinding wheel set-up, the cutter geometry set-up, the processing parameter set-up, the path calculation, and movement simulation and so on. Its control structure is shown in Fig. 1.

2.1 The Machine Tool Set-up

This module is used to set up the structure style of cutter grinders used (including the number of the coordinate axes and grinding wheel axes and their distribution form,etc.), structure parameters(the distribution of CNC axes, the machine coordinate system and the distance of travel of every coordinate axis, the distance of grinding wheel axes to the rotational center, the external dimension of the holding devices,etc.). All these are used to generated the coordinate transformation matrix for the interpolation of machine movement and the structure information of machines for the movement simulation.

Because of greatly diff-erent configuration forms of the tool and cutter grinder, the general method based on classification will lead to an extremely large system. After the careful study on the structure charactris-ics of tool and cutter grinders, the system is established with a generalized kinema-ics model, the necessary ma-ne information can be rapidly set up by the graphic menu, which make the system have a good universality

The kinematics model gene-rated consists of two movement transmission chains [1]. The branch from the cutter to the machine bed can be described as follows.

$$CT \xrightarrow{{}^{CT}_{1}Q} L(R)_1 \xrightarrow{{}^{1}_{2}Q} L(R)_2 \xrightarrow{{}^{2}_{3}Q} L(R)_3 \xrightarrow{{}^{3}_{4}Q} \cdots \xrightarrow{{}^{k-1}_{k}Q} L(R)_k \xrightarrow{{}^{k}_{M}Q} M$$

where CT is the cutter coordinate system; $L(R)_i$ is the i th axis movement matrix(L denotes translation, R denotes rotation); M is the machine bed; ${}^{i}_{j}Q$ is the position matrix of the i th movement axis coordinate system related to the j th movement axis coordinate system, so the known point r_t in the cutter coordinate system can be transferred into r_{mt} in the machine coordinate system as follows.

$$r_{mt} = {}^{k}_{M}Q \cdot [L_k|R_k] \cdots\cdots {}^{3}_{4}Q \cdot [L_3|R_3] \cdot {}^{2}_{3}Q \cdot [L_2|R_2] \cdot {}^{1}_{2}Q \cdot [L_1|R_1] \cdot {}^{CT}_{1}Q \cdot r_t \quad (1)$$

where $[L_i|R_i]$ means that either L_i or R_i can be selected. According to the same inference, the branch from the workpiece to the machine bed can be described as follows.

$$WP \xrightarrow{{}^{WP}_{n}Q} L(R)_n \xrightarrow{{}^{n}_{n-1}Q} L(R)_{n-1} \xrightarrow{{}^{n-1}_{n-2}Q} L(R)_{n-2} \xrightarrow{{}^{n-2}_{n-3}Q} \cdots \xrightarrow{{}^{k+2}_{k+1}Q} L(R)_{k+1} \xrightarrow{{}^{k+1}_{M}Q} M$$

where WP is the work piece coordinate system, n is the total number of the machine movement axis, ${}^{WP}_{n}Q$ is the position matrix of the workpiece coordinate system related to the nth movement axis coordinate system, so the known point r_p in the workpiece coordinate system can be transferred into the point r_{mp} in the machine coordinate system.

$$r_{mp} = {}^{k+1}_{M}Q[L_{k+1}|R_{k+1}] \cdots {}^{n-2}_{n-3}Q\ [L_{n-2}|R_{n-2}]\, {}^{n-1}_{n-2}Q\ [L_{n-1}|R_{n-1}] \cdot {}^{n}_{n-1}Q\ [L_n|R_n]\, {}^{WP}_{n}Q\ r_p \quad (2)$$

In the starting position of the process, r_p, r_t and R_i or L_i are all known and r_{mt} and r_{mp} are the same point in the machine coordinate system, i.e $r_{mt} = r_{mp}$, so numerical parameters of ${}^{i}_{j}Q$ in the formula (1) and (2) can be determined by the specific structure of the transmission.

2.2 Grinding Wheel Set-up

This module is demanded to have the cap- ability of receiving and setting up some paramet ers such as the number of grinding wheels insta- lled on the machine at the same time, the relati-ve position between grinding wheels, wheel ap-pearance , dimension, direction, grinding faces and their material.

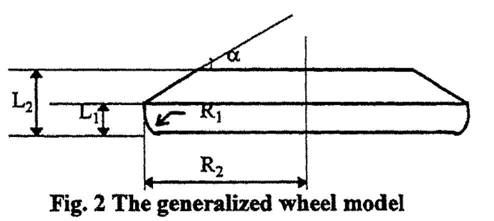


Fig. 2 The generalized wheel model

Then the description inform-ation of the grinding wheel group for setting up the process parameters, calculating the tool-path and simulating the movement process can be generated.

At present, the system provides ten kinds of grinding wheels such as the flat grinding wheel, the single bevel edge grinding wheel, the little angle single bevel edge grin-ding wheel, the cup-shape grinding wheel, the bowl-shape grinding wheel and the plate-shape grinding wheel. The information about grinding wheels are saved as the grinding wheel information file which make it can be modified and expanded conveniently. All parameters are input and selected by the graphic menu. The system displays the allocation state graph dynamically according to the information input. In order to simplify the design of the module for tool-path calculation and the module for movement simulation, all kinds of grinding wheels are transformed into a generalized description inside the computer(as shown in Fig.2).

.2.3 Cutter Geometry Set-up

This module is used to select the type of the cutters and input geometric structure parameters, in order to generate the cutter geometric information for calculating the tool-path and simulating the movement.

At present, the system can process cylindrical /tapered flat-end mills, cylindrical/tapered round angle mills, cylindrical/tapered ball-end mills, reamers and so on, and the system can be expanded conveniently.

Parameters set-up include the cutter diameter, cutter length, cutting edge length, the number of the peripheral cutting edges, helical direction and helical angle, rake angle of peripheral cutting edges, depth of the flute, first clearance angle of the peripheral cutting edge, second clearance angle of the peripheral cutting edge and structure style and parameters of the end cutting edge , etc..

When establishing the geometric model, all the available standards for current cutters have been collected and sorted out, which makes it convenient to program for standard cutters. On the other hand, some structure parameters can be defined by the consumer themselves for the requirement increasing the cutter performance. For example, consumers can define successively rake angle and depth of the flute of the cutting edge for ball-end mills by parametric equations or listed curves.

2.4 Machining Process Set-up

This module have the capability of providing the content to grind and the sequence of grinding, grinding wheels and work faces used at every grind step, position relation and parameters of grinding wheels and cutter faces, feedrates, etc..

On the base of analyzing and arranging the machining content and the machining process of every kind of cutters, a reference machining process base has been established in terms of different structure elements of cutters. When setting up a specific machining process, a reference machining process can be provided by the system according to the reference processing base, the set-up of the machine and the wheel. meanwhile, these reference machining process are permitted to be added, deleted and modified under the practicable situation. Thus the machining process set-up is more convenient and more flexible.

2.5 Movement Simulating

This function of this module is to display the relative movement and the grinding process cutters by dynamic 3D graphs according to the set-up information mentioned above. The display can verify the validity of programming data and the possibility of interference and collision between the wheel and the workpiece during grinding.

2.6 The Real-Time Control of The Grinding Movement

The movement path of wheels can be in real-time interpolated by using this module based on the parameters of geometry and grinding process, and the machine can be controlled by means of PLC instructions. The main processing steps are as follows.

a. The pre-process of the path. The machining process is divided into a series of steps which can be used for the interpolation device according to the processing path.

b. The interpolation of the movement path. The movement instructions of the every axis which meet the requirements for the processing feedrate are interpolated in real-time at the interruption period 8ms, then send to the position loop.

c. The control of increasing and decreasing feedrate. Increasing and decreasing feedrate are executed at the feedrate joint position, the starting point and the end point of the path.

d. The dynamic track display. The grinding movement path at the interruption period is displayed.

e. The detection and control of the system. The state of switches on the CNC operation panel and the machine.

3. THE PRINCIPLE OF GRINDING MOVEMENT SIMULATION

This simulation module is designed for mills, reamers , broaches, etc., the same feature of these tools are that they all have a complex contour. In the practical machining process, workpieces do rotation and translation, while grind wheel do rotation and swing. When we use dynamic images to show the machining process on the screen, not only the wheel but also the workpiece image need to be refurbished every times, hence the tradi-tional geometric modeling technique such as CSG model operation and swept volume model operation were not app-licable any longer[2]. Because of this, this paper presents a new method to partly change the original 3D problem to a 2D one. The method is based on the idea operating the line and surface instead of solid element as the basic unit. The detail of the algorithm is : read CL data at t_0 and subdivide wheel surfaces into triangles and trapeziums; then use a series of equidistant planes that perpendicular to the turning axis of the part as section to cut off the wheel and part(as shown in Figure 3), thus we can acquire two intersect curves generated by section with the wheel and part, the intersect curves are closed and expressed by polygon. Assuming the Figure 3 shown the machining state at time t_0, then a new closed approximate curve can be generated after subtracting the approximate curve of the workpiece at this location and the approximate curve of the wheel at the same one. The new curve is just the "finished contour" of position S_1 at time t_0. If we use three planes to do cutting at position S_1, S_2, S_3 (there are enough positions in practical sinulating), and the distance for each step is constant "Δd", then at those three positions we can acquire three finished contours expressed by polygon approximate curve, memorize these three curves, read the CL data at time t_1, repeat the operation that have been operated at time t_0 by using sufficient planes which follow position S_1, S_2, S_3 and along the moving direction of the wheel, note: the intersect curves of position S_1, S_2, S_3 should be calculated again if the positions are still within the machining zone at this moment, after all these have been done, memorize the new curves and update the old curves which have been changed. If the step size of position S_1, S_2, S_3 …… is Δd and the length of part is L, thus we can acquire n finished polygons($n=L/\Delta d$). A lamella can be composed of two adjacent curves, as shown in Figure 4, we call this lamella "slice unit". When used for displaying, slice unit will been drawn pieces by pieces from far to near according to the observer's direction, this method significantly simplified the visual surface determination, because the sightless surface will be automatically covered by the image drawn later, the only problem remained is the visual surface determination of the slice unit itself. The another advantage of this algorithm is: we can avoid doing a large number of repeated calculation when

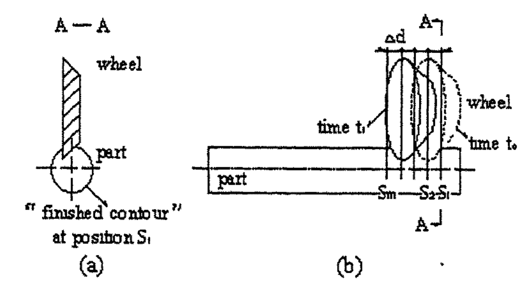


Fig.3 (a) Finished contour generated by wheel and part at s_1 (b) machining state at time t_0 and t_1

changing the visual angle, because the data acquired before can be reused after multiplied by a transformation matrix. In this algorithm, the calculate and display precision can be controlled respectively without affecting each other.

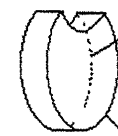


Fig.4 A slice unit is composed of two curves

The model used for interference detection in this simulation system is a hierarchy of Oct-sphere model[3], the interference detection is in step with cartoon's operation, once the interference and collision occur, stop the operation immediately, report where and when the interference and collision occurred. Before the verification begin, the moving bodies own models must have been constructed, then using move equation to count each son node's position, thus can acquire the collision and interference state by seeking the Octree. The improving that hierarchy of Oct-sphere model has is that not like the Octree, it use ball at each son node instead of using cube, so if we want to know whether the two node have collided, we only need to judge the distance of the two ball's center.

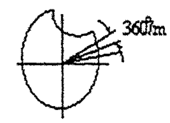


Fig.5 Divided curve into m portions

As a integrated simulation system must have the precision detect function, this system detect the precision by comparing discrete point. Accordingly, we have the finished curve of position S_1 expressed by a polygon, then set the intersection which generated by the cutting plane of position S_1 and X axis being the center of a circle, as in the Figure 5, equally divide the polygon into the m portions in the 360 degrees, evaluate the coordinate of each point which both at the bisectrix and polygon; using the same method to obtain the coordinate of ideal product, then make the machining result and ideal result do subtraction operate, presume that is the error of each point, the position S_1 is qualified if $\max|\delta_{xyz}|$ satisfy the precision. The detect precision could be improved by increasing m and decreasing the step of position $S_1, S_2 \cdots\cdots$.

4. THE PRINCIPLE AND ALGORITHM OF THE REAL-TIME INTERPOLATION FOR THE GRINDING MOVEMENT

On the base of analyzing and arranging the geometric structure and machining process and methods of every kinds of cutters. All structure elements to be machined are divided into a series of sub-blocks for calculating the tool-path, which can compose basic function units of interpolation devices. Every sub-block expressed in the form of parameters is used to grind cutters structure elements with the same mathematics model and the grinding process.

The collection of sub-blocks can fulfill all process content in the reference machining process base. Combining different, the system can complete the whole grinding process for this cutter. The analysis taking rear face of balll-end mills for example is shown as follows.

4.1 The Path Expression for the Movement of Grinding Rear Face of Ball-end Mill

In Fig.5, **oxyz** is the workpiece coordinate system whose origin point coincides with the ball center. For the helical line of the cutting edge with either the constant lead or constant helical angle, the curve can be expressed as $r_p = r(\theta)$. It is supposed the rear face is defined in the normal section, the clearance is $\alpha(\theta)$. If $n(\theta)$ is the unit normal vector of the rear face at the point $r(\theta)$, $a(\theta)$ is the unit tangent vector of the cutting edge, then

$$\begin{cases} a(\theta) = [dr(\theta) / d\theta] / |dr(\theta) / d\theta| \\ n(\theta) = \cos\alpha(\theta) \cdot r(\theta) / |r(\theta)| + \sin\alpha(\theta) \cdot a(\theta) \times r(\theta) / |r(\theta)| \end{cases}$$

writing $v(\theta) = a(\theta) \times n(\theta)$, then in the local coordinate system $(v(\theta), a(\theta), n(\theta))$, the wheel path is:

$$\begin{cases} \boldsymbol{u}_p(\theta) = \sin\delta(\theta)\cdot\boldsymbol{a}(\theta) + \cos\delta(\theta)\cdot\boldsymbol{n}(\theta) \\ \boldsymbol{r}_p(\theta) = \boldsymbol{r}(\theta) + R_1\boldsymbol{n}(\theta) + (R_2 - R_1)\dfrac{\boldsymbol{n}(\theta) - (\boldsymbol{n}(\theta)\cdot\boldsymbol{u}_p(\theta))\boldsymbol{u}_p(\theta)}{\left|\boldsymbol{n}(\theta) - (\boldsymbol{n}(\theta)\cdot\boldsymbol{u}_p(\theta))\boldsymbol{u}_p(\theta)\right|} - R_1\boldsymbol{u}_p(\theta) \end{cases}$$

where $\delta(\theta)$ is the grinding angle. By the aid of the coordinate matrix obtained from the machine setting-up module, the movement path of every coordinate axis of the machine (translation axis x, y, z, rotation axis A, B), $[\boldsymbol{r}_m(\theta)(x_m(\theta), y_m(\theta), z_m(\theta)), \phi_A(\theta), \phi_B(\theta)]$, can be gained.

4.2 The Real-time Interpolation Control for the Grinding Movement

The system is a closed-up control CNC system and the digital sampling interpolation method based on time subdividing is adopted. The movement components of every axis at the (K+1)th period are computed at the Kth work period in real-time, and it should meet the assigned feedrate requirement. The calculation steps are:

(1) Determination of the feedrate. The feedrate $F(t_k)$ at the interpolation period is determined by some parameters such as the programming feedrate, the adjusting coefficient for the feedrate on the control panel, the limitation of movement of every coordinate axis and the capability of the servo system and so on. It is required to assure the constant feedrate grinding of the wheel relative to the cutter ground and the smooth transition of the feedrate.

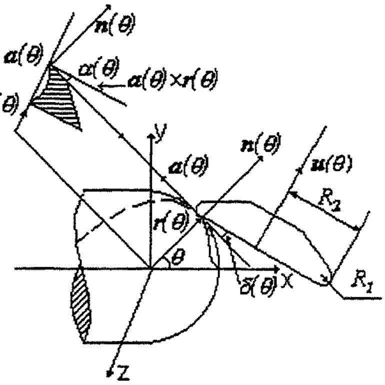


Fig. 5 The movement of grinding rear face of ball-end mill

(2) Determination of the feed step. If the interpolation period is T, then $f(t_k) = F(t_k)\cdot T$

(3) Calculation of the movement increment for every axis.

$$\begin{cases} \Delta\theta = f(t_k)/\left|dr(\theta_{t_k})/d\theta\right| \\ \theta_{t_{k+1}} = \theta_{t_k} + \Delta\theta \\ \Delta\boldsymbol{r}_m(t_{k+1}) = \boldsymbol{r}_m(\theta_{t_{k+1}}) - \boldsymbol{r}_m(\theta_{t_k}) \\ \Delta\phi_A(t_{k+1}) = \phi_A(\theta_{t_{k+1}}) - \phi_A(\theta_{t_k}) \\ \Delta\phi_B(t_{k+1}) = \phi_B(\theta_{t_{k+1}}) - \phi_B(\theta_{t_k}) \end{cases}$$

By sending the above movement increments to the position loop, controlling the grinding movement can be achieved.

5. REFERENCE

1. A. Ruegg, "A Generalized Kinematics Model for Three-to Five-Axis Milling Machines and Their Implementation in a CNC", Annals of the CIRP Vol. 41, No.1, pp.547-549,1992.

2. Hook T V. "Real-Time Shaded NC Milling Display". Computer Graphics, Vol.20, No.4, pp.15-20, 1986.

3. D W Sandberg, "Collision Detcetion Using Sphere Approximation", Proc. Int. Conf. on Robotics and Factories of Future, , pp.457-460, California, 1987.

A GENERAL IMPLEMENTABLE APPROACH TO DISTRIBUTED PROBLEM SOLVING SYSTEM

Baogang Wei, Zheng Qin, Jichang Kang and Huacan He
Department of Computer Science and Engineerng,
Northwestern Polytechnical University
Xi'an, 710072, P.R. CHINA

ABSTRACT

Cooperative problem solving is an important paradigm for the next generation of industrial intelligent systems. One of key problem to use it in engineering domain is development of a structured design method. In this paper, we put forward a design method of distributed artificial intelligence (DAI) system based on software engineering, describe the detailed design process of a DAI system through an example of a simulative transformer substation system, and present some of key problems and techniques of DAI in engineering domain, such as system modeling, task decomposition and allocation, cooperative mechanism, and so on.

KEYWORDS

Distributed Artificial Intelligence, Cooperative Problem Solving, Software Engineering, Simulative Transformer Substation.

1. INTRODUCE

With the gradual acceptance of artificial intelligence techniques, more and more sophisticated systems have and will be developed. They are used to supervise and control manufacturing processes, power networks, chemical plants, and so on. However as the system's components become more complicated, it is difficult to cooperate these components.

The traditional AI development methods are insufficient to create effective cooperative systems. However, radical improvements in communication and network have increased the advantages of distributed applications. A distributed approach offers certain advantages for complex problem solving: faster response, increased flexibility, robustness, resource sharing, graceful degradation, and better adaptability. The main reasons for developing such systems are geographic distribution in the domain of applications, controlling the increasing complexity of AI systems, and increasing the power of resulting systems. Even though researches of DAI have obtained many of gratifying achievements, such as system modeling, coherency, communication among agents, and so on, most of them have been in experimental and theoretical stage. There are many problems must be solved so as to use them in engineering domain. One of the main limitations is the lack of methods to structure the development process [1].

This article explores a development approach to DAI system in engineering domain, called software engineering-based design method. In section 2, the design principle and steps of DAI system are described, and in section 3, an example of a simulative transformer substation system is given to show how to use the approach in design process of the system, and a new multiagent cooperative method, called two-level-priority-based cooperative method, is presented. The interactions between agents are classified as two levels: the static level and the dynamic level. The priorities of the static level's interactions are predetermined statically, meanwhile the priorities of the dynamic level's interactions are dynamically determined in process of problem solving.

2. SOFTWARE ENGINEERING-BASED DESIGN METHOD OF DAI SYSTEM

Most of current DAI application systems are very large and complex. Such a complex system's design is a iterative and improved process. Summarily, there should be following several main steps:

• requirement analysis
• determination of system overall composition
• task decomposition and allocation
• the decision on a suitable cooperation mechanism and problem solving strategies

Requirement analysis: Requirement analysis determines what function and performance the system will bring about, what information is to be processed, what interfaces are to be established, what design constraints exist, and so on.

Determination of system overall composition: After analyzing system's requirements, the system overall composition can be determined. This bases mainly on system functions, environment constraints, as well as system's input and output features. Among of these factors, the system's input and output are the most important aspects, because any software system can be thought to be a information processing system and both information processed and generated by system determine the system's features to a great extent.

Task decomposition and allocation: System complexity is reduced by decomposing the task into modules. The rules of decomposition are to make models have strong internal cohesion and low coupling with each other. That is, the objective of coupling has two dimension: minimizing the connection between modules, and maximizing the relationship among elements within the same module.

The decision on a suitable cooperation mechanism and problem solving strategies. A DAI system is composed of a group of loosely-coupled intelligent agents. The capacity of each agent in the system is limited by its computing resource, its knowledge, and its perspective, so that anyone cannot solve a overall problem lonely. There must be some form of interaction among agents in order to reach a coherent situation. Therefore the researches of cooperation mechanism and problem solving strategies are very important.

Generally, the architecture of agent is consisted two layers [2]. One is problem solving level, and the other is cooperation layer. Each layer has several modules with different functions (see Fig. 1).

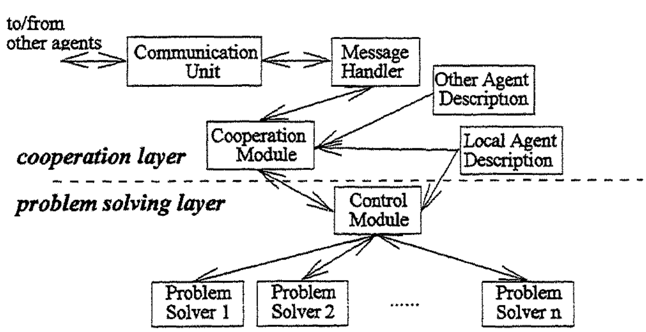


Fig. 1: Architecture of an agent

The cooperation layer's objective is to ensure that agents domain level activities are coordinated with those of others. Other Agent Description and Local Agent Description are the cooperation layer's two main components. Cooperation Module uses them to decide which tasks should be performed locally, and determine when social activity is appropriate. Message Handler and Communication Unit are used to receive or send information to other agents. The problem solving layer has a Control module and some of problem solving modules. The Control module allocates tasks among problem solving models. Each problem solving module has its own knowledge-base and inference engine and can solve a special domain problem.

Researchers have developed a range of cooperation methods for distributed problem solving or multiagent systems. Edmund H. Durfee [3] and his colleagues summarized six main cooperative approaches including compiled conventions and social laws, agent/module specification, organization structure, distributed planning/scheduling, contracting, and observation. Each of them is applicable to a restricted set of domain. No single approach appears to satisfy all needs.

3. AN EXAMPLE OF SIMULATIVE TRANSFORMER SUBSTATION

Transformer substation is an important department of electric power system. The operators of such system must be capable of managing the system both under normal conditions and in the presence of system malfunctions. Their ability to diagnose faults and take appropriate corrective actions promptly is highly desirable [4]. But the complex and dangerous features of a transformer substation make it impossible to train operators on real equipment. Therefore developing a simulative transformer substation is a effective way to train operators. Building such a system deals with a set of complex tasks, such as the control of input and output devices, load flow calculation, diagnosis of faults and abnormal events, and so on. Therefore an integrated and distributed problem solving architecture seems to be a good choice.

3.1 Requirement Analysis

The aim to develop the simulative transformer substation is to improve operator's operation skill and his faults' and abnormal events' diagnosis ability. Generally, the system should include the following functions:

- to collect switch status and supervise network's change
- to identify and diagnose faults and abnormal events
- to calculate load flow under both fault and normal conditions
- to display the electric network status through computer graphic interface and meters in real-time
- to have a friendly user interface so that instructor can set up fault and abnormal events
- to simulate various fault and abnormal phenomena

3.2 System Overall Composition

The simulative transformer substation is a large complex system which combines physical simulation with computer digital simulation. Therefore system model is based on not only its overall performance, but also the environment constrains. For example, input and output of system should reflect their change of status logic, that is, system must be of ability of real-time input and output. In addition, in order to facilitate the instructor to set up faults and abnormal events, there must be a user interface agent. It presents the teacher with a graphical display representing the electric network status. It is mouse-driven and uses pull-down menus for ease of use. The fault diagnosis task which is time consuming one should be performed by a special agent. Therefore the system is composed of three main agents: Network monitor, User interface and Fault diagnosis (see Fig. 2).

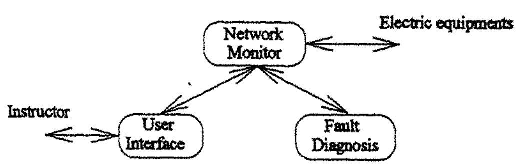


Fig. 2: System overall composition

3.3 Task Decomposition and Allocation

In above system module, each agent performs a class of performance. Network monitor agent is responsible for the real-time tasks such as system's input and output, network calculation, and fault simulation. User interface agent facilities instructor to interact with system. Fault diagnosing agent deals with complex and time consuming diagnosis task.

After determining system's overall composition, some of tasks can be divided into smaller subtasks. For example, there are two types of equipment in the simulative transformer substation. One is controller and the other is protective relay. Data collection on them can process independently. The fault diagnosis can also be divided two subtasks, one based on network status and the other on protective logic. The relationships of tasks are shown in Fig. 3.

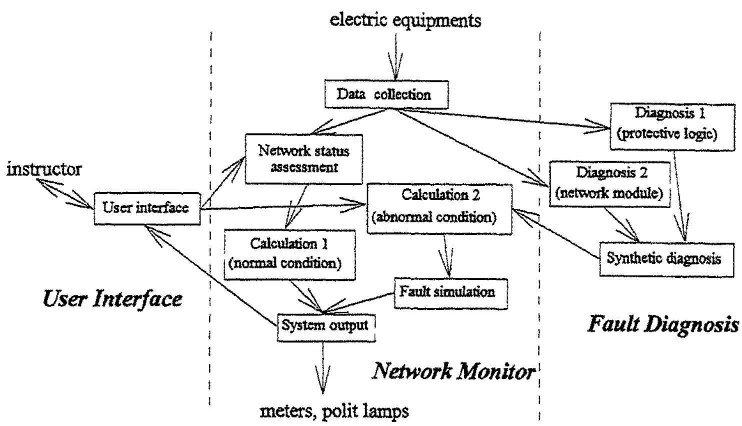


Fig. 3: The relationships of tasks

3.4 Cooperation Mechanism and Problem Solving Strategy

The cooperation among autonomous intelligent agents is the key problem of DAI. A variety of cooperation approaches for DAI have been put forward and studied. They have ranged from centralized approaches to negotiation strategies [5] or from very statically-defined models to models that can dynamically change with environments. Each of them is applicable to a restricted set of domain. No single approach appears to satisfy all needs. In engineering application domain, what method is chosen to achieve the balance between stability and flexibility is very important.

We adopt a two-level-priority-based asynchronous communication approach to realize the interactions among agents in the system. The approach, which is used in such complex real-time system, is more efficient than other types of DAI cooperative methods such as blackboard system or contract net. For the interactions with respect to different functions, we statically define their priorities according to their important extents. For example, the priority of diagnosis message which Monitor Agent receives from Diagnosis Agent is prior to the one of set message from User Interface Agent. While for the interactions which belong to a certain function and change with the environment, their priorities are calculated dynamically according to their urgent extents. For example, when multiple faults occur simultaneously, since one fault under different environment conditions has different urgent extents, and it is difficult for fault simulation module to determine its order in the fault set, therefore, this dynamic process, which deals with a lot of complex knowledge about logical relationship between environment condition and fault, is processed by fault diagnosis module is more suitable than by fault simulation module. We call the former first level priority and the latter second level priority.

Agents in the cooperative problem solving system interact to a large extent by exchange of messages. In the system, the messages exchanged among agents are listed in table 1. At any point of operation, an agent might have several incoming messages in its handler to be executed. Generally, there tow disjoint types of messages, one is strictly used in communicative acts for initiating actions, and the other is used in response to former acts. Selecting which one to execute first is important for cooperative problem solving. In the system, it is according to message's two level priorities.

Table 1: Messages exchanged between agents

source agent	goal aent	message content
Network Monitor	User Interface	control signal, protective signal, load flow, reject set
	Fault Diagnosis	control signal, protective signal
User Interface	Network Monitor	electric network set, fault and abnormal set
Fault Diagnosis	Network Monitor	result of diagnosis

4. CONCLUSION

A major issue of DAI used in engineering domain is the lack of development approach. To solve this problem, we introduce a software engineering-based design method of DAI system, and describe in detail its implementable steps: requirement analysis, determination of system overall composition, task decomposition and allocation, and decision on a suitable coordination regime and problem solving strategy. What is required to stress is the order of the four steps is not unchanged. They affect and restrict mutually. A new multiagent cooperative method, called two-level-priority-based cooperative method, is presented. It not only satisfies the system's real-time requirement, but also has certain ability to dynamically respond to environment change.

5. REFERENCES

1.George P. Lekkas, Nicholas M. Avouris, and George K. Papakonstantinou, "Development of Distributed Problem Solving System for Dynamic Environments", IEEE Transactions on System, Man, and Cybernetics, Vol.25, No.3, pp.400-414,1995

2.N. R. Jennings, "Controlling Cooperative Problem Solving in Multi-agent Systems Using Joint Intentions", Artificial Intelligence, Vol.75,No.2, pp.195-240,1995

3.Edmund H. Durfee, Daniel Damouth, Marcus Huber, Thamas A. Montgomery, and Sandip Sen, "The Search for Coordination: Knowledge-Guided Abstraction and Search in A Hierarchical Behavior Space", Proceeding in 4th European Workshop on Modeling Autonomous Agent in a Mult-Agent World, pp.164-183,1992

4.T. Minakawa, Y.Ichikawa, M.Kunugi, K.Shimada, N.Wada, M.Utsunowiya, "Development and Implementation of A Power System Fault Diagnosis", Expert System, IEEE Transactions on Power Systems, Vol. 10, No. 2, pp. 932-940,1995,

5.Nicholas V. Findler and Gregorg D. Elder, "Multiagent Coordination and Cooperation in A Distributed Dynamic Environment with limited resources", Artificial Intelligence in Engineering, Vol.9, No.1, pp.229-238,1995

A NEW APPROACH TO SOFTWARE DEVELOPMENT IN CONCURRENT ENGINEERING

Jianxin Yan, Chungfan Mai, Tang Ming
Dept. of Computer Sci. & Eng.,
Beijing Univ. of Aero. & Astro.
Beijing 100083, P.R.China

ABSTRACT

This paper presents a new approach to software development in Concurrent Engineering(CE), the concurrent engineering approach to software development, which originates from the basic thoughts of CE and can increase the efficiency of software development. Our approach is a kind of 'CE for CE'.

KEYWORDS

concurrent engineering, software concurrent engineering, software development methodology

1. INTRODUCTION

Concurrent Engineering(CE) practice will not success without changes in management structure and culture in industry. Technology alone will not make it, either. When furthering CE research and practice, however, computer-based information system(including platform and tools) utilization is the most important factor that promote CE.

Borrowing the thoughts of concurrent engineering(CE), we establish the CE approach to software development, i.e., software concurrent engineering(SCE) approach. Instead of doing all of requirements analysis, followed by all of designs, ..., --traditional software development methodologies requires to do so--, SCE suggests that one does analysis where it is appropriate, design where it is appropriate, ... That is to say, SCE advocates to continue all follow-up development activities once enough (NOT ALL) information is available. Thus, with SCE's cooperative team work, requirements analysis, design, coding and testing activity will be performed as concurrently as possible to shorten the software life cycle. Additionally, effective user participation and teamwork mode enable all members of a project group to cooperate closely and communicate promptly in the SCE development process, which contributes to eliminate redesigns, reduce cost and enhance software quality.

Our SCE approach can conduce to enhance information system utilization in CE.

2. SOFTWARE CONCURRENT ENGINEERING: STRATEGY AND FEASIBILITY

Traditional software development, i.e., waterfall-style development, is usually a sequential process, which is a series of discrete phases such as requirements analysis, preliminary design, detailed design, coding and unit testing, integration and system testing. The waterfall-style development usually implies that all requirements analysis is completed before going on to design, and that all design is completed before coding starts, though Royce recognized that software development is not simply a series of discrete activities, strictly but a set of iterative activities[7]. In the early research on software development, it was clearly understood from the waterfall model that the earlier an error is made in a project the more catastrophic the effects of that error. So, the early software engineering stressed that review must be held in the end of each phase to eliminate all errors, and that the order of the development phases could not be broken, only when a change would be introduced, one could return to some previously completed phase. Having too many deficiencies which may have adverse effects on software projects, the waterfall-style development has been attacked by many authors[12].

Since the middle of 1980's, rapid prototyping approach, instead of waterfall-style development, has been the mainstream method of software development. Recognizing that software development, particularly during its early stages, is a learning process, rapid prototyping approach provides us both a strategy: admission of failure, and a philosophy: feedback and adjusting, i.e., step refining[10]. Speaking rigidly, prototyping is just such a thought which doesn't provide either a uniform norm or a general methodological procedure. Just According to the relationship between a prototype and the final system, the approaches to prototyping may be divided into three categories: throw-it-away prototyping, evolutionary prototyping and incremental prototyping.

Spiral model is one of the most famous prototyping methods, it shows software development as a gradually expanding spiral, each cycle of which begins with the identification of objectives, then do risk analysis and prototype development, and is completed by a review with go/no-go plans made for the next cycle. After each cycle, a middle product, i.e., prototype, is produced. The last cycle produces the software product[2]. Though the spiral development is not a traditional sequential process, each cycle of the spiral, where the waterfall model or its clones is usually adopted, is almost still a linear process.

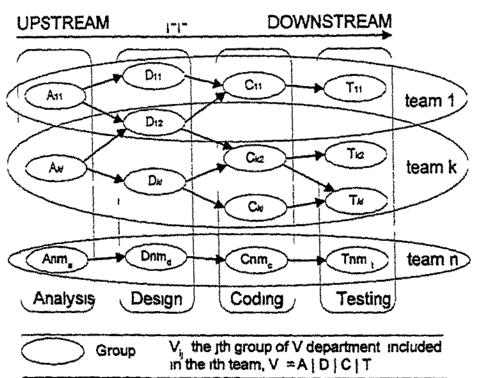


Figure 1 SCE-style Development: Function Groups and SCE Teams

Software concurrent engineering is the concurrent engineering approach to software development, it aims to shorten the software life cycle, reduce cost and enhance the software quality. SCE-style software development, which stresses to organize the existed linear and iterative development activities as concurrently as possible, is a concurrent process based on team work. We may decompose development activities such as requirements analysis, design, coding and testing, which were organized in different phases, into some sets of sub-activities, then assign them to the multifunctional developing teams, each of which includes analyst, designer, programmer and testing engineer, etc. (Figure 1) SCE breaks the phase constraint thoroughly. Phase transitions triggered with the delivery of a set of documentation will be replaced by transitions from upstream sub-activities to downstream sub-activities, so engineers engaged in different jobs can communicate flexibly and promptly. In a macroscopic sense, different development phases will be performed concurrently. In fact, concurrent development has existed in industrial practices --people recognized that the earlier the precise designs of components of a system are completed, the earlier the total system approaches the reasonable design successfully--, but some failed. On the other hand, software engineering requires strict project management, and it's commonly regarded that the linear development guided by the waterfall model conduces to manager and control the development process, so, even though in the recent years concurrent development has been advocated by some people, for example, G. Booch encouraged "design a little, coding a little" in Ada software development[3]; the fountain model[6] raised by W. Henderson-Sellers, et al. also supports the concurrent development, because there are conflicts between concurrent development and old concepts and managerial pattern in software engineering, and no methodology with detail operating regulations is provided, concurrent development practice has never been popular. Considering the reasons listed below, we argue that SCE practice supported by appropriate methodologies is worthy to be advocated:

1). Software development is a process where approximate solution is found

Absolutely right and complete requirements hardly exist. Even though the software has been developed to satisfy current user meets, the system requirements may change constantly for ever. So software development is always a process to find approximate solution with imprecise requirements. We usually draw on the "ideal requirements" in initial phase of software life cycle, but we certainly can approach the "ideal requirements" during the development process.

2). Characteristics of software

Anyone who has worked on a project of significant size knows that requirements are not all at the same level of abstraction. Some are broad and high-level, others are very detailed. In addition, different parts of requirements have different stability. Some are constant, easy to be defined and frozen, and others are instant and difficult to be frozen. This means that, once the entirety of requirements is examined, some requirements may be deferred until a late time. After the constant part of requirements is defined, while trying to define the instant part, we may do design for the defined requirements. Likewise, some design, coding and testing concerns can be done in advance, others can be deferred until later.

Additionally, software can be easily divided into independent components, which also make it possible to develop software concurrently.

3). With the development of software reuse, software development doesn't start from scratch again. In many projects, engineers now needn't write all codes of a system, some parts of which may be integrated and tested just after their requirements are defined through reusing the existed codes -- obviously, it is unnecessary to defer this job.

4). While changing the organization and culture of the project group, especially, changing the working habit of manager and reviewer, we can ensure the development process controllably.

Additionally, with the fast development of network technology, it's possible to develop a software concurrently for a project group whose members are scattered over different geographic areas.

Speaking broadly, prototyping-style development is a special case of SCE; it is a conservative SCE practice. For example, when the total life cycle is observed, the evolutionary prototyping approach also makes requirements analysis, design, coding and testing done overlappingly (or concurrently in a macroscopic sense). However, in the evolutionary prototyping life cycle, not only several prototypes can be developed simultaneously, each prototype itself can also be developed concurrently.

3. PROCEDURE OF THE SCE APPROACH

According to [5], using the terms of the American DoD-2167A Standard[9], software development is usually orgazined as the development of one or many CSCIs (computer software configration item), which is usually one part and sometimes all of the software under development. Once development activities start, there are four types of activities with which personnel may be involved: defining requirements specification for CSCI, doing preliminary design(PD) for CSCI, developing the subordinates of the CSCI, or performing system testing of the CSCI .

When performing PD for the CSCI, engineers do all the usual activities which PD includes. They define the functions, inputs, outputs, etc., of the CSCI, then they decompose it into its subordinates, i.e., computer software component(CSC), and they repeat the process recursively for each subordinate. PD of the CSCI must complete the design of its all subordinates, i.e., CSC_i.

Identifying the i subordinates, which comprise the CSCI, launches the i processes of developing CSC. Similarly, when developing each CSC_i, engineers must be involved in one of the three types activities: designing CSC_i, developing the subordinates of CSC_i, i.e., CSC_{ij}, or performing integration and testing for CSC_i. As design continues, new CSCs are created, for each a developing process is launched. This continues until we finally arrive at the level just prior to the creation of computer software units(CSU). At this level we perform detailed design of $CSC_{ij...m}$s. Finally, we develop CSUs that are spawned by those $CSC_{ij...m}$s. In fact, coding and unit testing are the main tasks of the CSUs development.

So, the traditional software development procedure can be described as:

1) *completing requirements analysis for all CSCIs to define a detailed software requirement specification.*
2) *do the designs of all CSCIs, all CSC_i s, all CSC_{ij}s, ..., up to the completion of the design of all CSUs.*
3) *coding and unit-testing CSUs*
4) *integrating and testing CSUs, ..., CSC_{ij}s, CSC_is, then do software system testing of CSCIs to finish the developing process.*

Ideal SCE-style development is a kind of recursive-decomposition-and-parallel-development, we should:

1) *systematically decompose the problem into highly-independent CSCIs.*
2) *re-apply the decomposition process to each of CSCIs to decompose them further. (recursive decomposition: CSCI ⇨ CSC_i ⇨$_j$-⇨CSU)*
3) *accomplish this re-application of the process simultaneously on each of the components.*

It's certainly impossible to perform the ideal SCE development perfectly in practical software development. However, we can abide by the rule: in the development process, once enough information is available, we continue the next activity. Transition from upstream to downstream activity needn't be waited. That is to say, if requirements of a component are defined, we continue do design of it, we needn't wait until total requirements of the system is completed. Thus, analysis, design, coding and testing are decomposed into sets of sub-activities, which can be performing concurrently. Please remember: each sequence composed of analysis, design, coding and testing atomic activities, i.e., sub-activities not decomposed again, is usually performed linearly and where iteration is permitted. This means that the intrinsic order cannot be broken, that is to say, it's impossible to for software development to do testing first, then do design and coding, and define requirements at last. So you can say what SCE stresses is the optimization of the working procedure. Practical SCE steps may be:

1) *defining the overall requirements of the software, then executing step 2,3,4 repeatedly until the concurrent developing process is fully launched with the expected concurrent level.*
2) *decomposing, refining the system requirements to identify and isolate CSCI.*
3) *developing the isolated CSCI and its subordinates, to identify CSC_i/.../CSU and start the development process of CSC_i/.../CSU.*
4) *performing unit testing for the CSUs whose coding is finished, and performing integration and testing for the CSUs that pass the unit testing.*
5) *performing system integration and testing.*

According to both the quality of the development teams and the specified situation of the project, such as the difficulty, schedule, etc., you can flexibly determine the time when concurrency occurs and the concurrent degree of the development process.

4. KEY ELEMENTS OF SCE

In this section, we discuss the key elements that ensure SCE practice successful.

4.1 Changing Philosophy And Material Foundation

SCE is first a new culture/philosophy, then a software development method. First of all, it is necessary to imbue all members of the project group with the thoughts and concepts of CE. To make SCE more effective, you'd better be equipped with a distributed software engineering platform and some communication tools, which can ensure that everyone gets whatever information on the developing software he wants at any time.

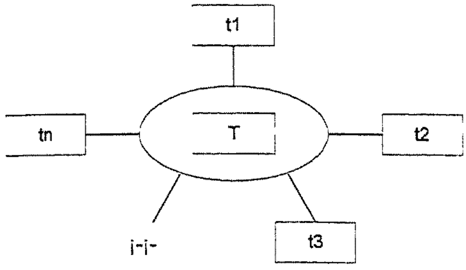


Figure 2. Organization of SCE Development Teams

4.2 Orchestrating Project Organization And Management

In SCE practices, organization development teams and synergy among human and material resources is one of the most important key elements. SCE to some extend improves the yielding efficiency just through increasing the input of human, material and financial resources per unit time.

In SCE practices, developing teams may be organized as Figure 2, where t_1, t_2, ..., t_n(n>=2, the value of n depends on the factors such as the expected concurrent level of the developing process, etc.) are multifunctional developing teams; T is the leader team, whose members include project manager, the leader of each team(t_1, ..., t_n), and user representative(s); the ellipse area represents the shared perception (view) of the project. Under the leadership of project manager, T team is responsible for not only developing processed allowing for the downstream processes from the initial stages of development, but also orchestrating the project organization and management

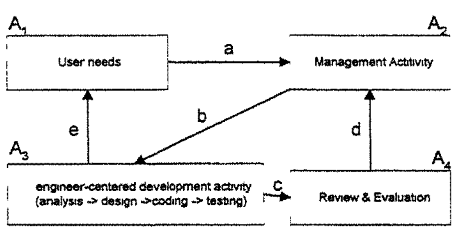


Figure 3. Interaction among User, Engineer,Manager and Reviewer

Figure 3 describes the interactive mode among engineers, management personnel, reviewers and users. A_2 decides the project plan and orchestrates the overall order of all other activities. New and/or changing user needs drive the spawning of new and the modification of old project plans($A_1 \Rightarrow A_2$ via arrow *a*). The project plan drives the initiation of engineer-centered development activities($A_2 \Rightarrow A_3$ via arrow *b*). The completion of various intermediate products triggers the initiation of specific reviews($A_3 \Rightarrow A_4$ via arrow *c*). The passing or failure of any review signals the project plan ($A_4 \Rightarrow A_2$ via arrow *development*) to proceed with appropriate commands to A_3 (arrow *b*). Finally, the creation of software solutions by A_3 changes the state of user needs satisfaction(arrow *e*).

Unimpeded communicating channels must be maintained among all members of the project group to ensure smooth transitions between upstream and downstream sub-activities. Management and review personnel must be under the state of either working or order-pending at any time. They should promptly response to requests coming from the project group in a way like interrupt in hardware devices.

4.3 Decomposition of Software Components And Early Isolation of Stable Requirements

To make the concurrent developing process work, the rules listed below have to be met:

1) *Delaying requirements freeze.*
2) *The entirety of the information available for the software project must be examined to carefully determine which decision can safely be deferred until later.*

3) *The system components and their interfaces must be well defined, and kept constant*
4) *The components of the software system must be loosely coupled and highly cohesive.*
5) *The criteria for deciding to stop decomposition must be made so that the concurrent developing process can be controllablely convergent.*

4.4 User Participation All Over the Life Cycle

Effective user participation is one of the key factors that ensure it possible to develop a software successfully. The "bucket theory" in system engineering tells us that the capacity of a bucket, which is composed of the planks with various lengths, is decided by the shortest plank. Obviously, it is impossible to solve thoroughly the problems underlying the software development then increase the efficiency of software development just by perfecting the technologies which software engineers hold. Laying more stress on the side of software engineers than on the side of users in software engineering research is one of the important points which result in the expensive, endless, low-yielding developing process.

Since prototyping technology emerged, software engineering advocates that users at least participate the activities such as requirements analysis, test of software components and prototypes, validation and evaluation of prototypes. SCE divides the various developing phases into concurrent sub-activities, .which objectively enhances the level of user participation. In our SCE practices, additionally, we initiate a new kind of user training activity, which is held in the intervals between the "busy" periods when users take part in the developing activities, by software engineers to imbue users with some relevant computer knowledge, the contents, difficulty and training schedule of which should be carefully arranged, in order to enhance the users' quality on computer technology gradually. Though this kind training activities maybe doesn't produce instant results, it will benefit both the developing teams and the users in the phase of software maintenance, and in the later projects for the long-lasting cooperating companions. Especially, it is strategic for improving the whole software industry.

4.5 Software Quality Assurance(SQA) Penetrating the Overall Life Cycle

In the life cycle of SCE-style development, many activities of requirements analysis, design, coding and testing are done overlappingly. Even in the end of the development process, there still exist the activities of design, and even requirements analysis. So, SQA must be penetrated all over the life cycle.

In short, risks of one project come from two sources: 1)there are vague part and/or misunderstanding of user requirements; 2) engineers draw technical mistakes into the developing system. SCE demands to commit users to participate every stage of the life cycle, so that it can eliminate the risks introduced from requirements to the most possible extend. On the other hand, adopting the common methods such as review, verification and validation, SCE encourages to try and evaluate the alternative solutions in order to select the best solution, which may be a hybrid solution, though innovative approaches are discarded in favor of conservative technologies in the traditional software engineering practices. SCE also advocates to reuse the matured products of customized components to avoid introducing mistakes.

5. SCE & CLASSICAL SOFTWARE DEVELOPMENT METHODOLOGIES

Concurrent development advocated by SCE is independent of software development methodologies, in other words, no matter what methodology, either structured development or object-oriented development, is feasible to be adopted for carrying out SCE-style development. Furthermore, our research and practice show that object-oriented development is more suitable for SCE practice than structured development[11]. Firstly, object model establishes a more constant model for a software system than functional model does. At the same time, object model directly maps the application domain into the solution domain, so object-oriented analysis makes it easier to identify, track and evaluate the influence produced by changes in the system under the development[4]. Secondly, object-orientation's intrinsic mechanisms such as abstraction, encapsulation, inheritance (classification) and polymorphism conducing to increase cohesion, reduce coupling and eliminate redundancy, object-oriented development can reduce system complexity and enhance system extensibility[8]. Additionally, the consistency of object model not only ensures the smooth transition from requirements analysis through design to coding, but also conduces to enhance cooperation among analyst, designer, programmer and testing engineer.

6. CONCLUSION

SCE approach is a new approach to software development and a kind of 'CE for CE'. SCE-style software development is a concurrent process based on team work. It can conduce to enhance information system utilization in CE and do help for various software system development.

At present, we adopt Lotus Notes as the facility to orchestrating the cooperation and communication among the SCE development teams. However, to make SCE give full play to its advantages, we should go into the further level of SCE, or computer supported software concurrent engineering(CSSCE), where we must try to exploit SCE supporting technologies and develop its tools, build a CSSCE environment.

7. REFERENCE

1. *AD-A225128, Final Report: DAAPA Concurrent Design/Concurrent Engineering Workshop, Key West. Florida, Dec,6-8, 1988*
2. *B. Boehm, "A Spiral Model of Software Development and Enhancement", IEEE Computer, 21, 5(May 1988), p297-311*
3. *G. Booch, "Describing Software Design in Ada", ACM SIGPLAN Notices, Vol. 16, No.9, Sept. 1991, p42-47*
4. *P. Coad, et al., Object-Oriented Analysis, Yourdon Press, 1990*
5. *A. Davis et al., "A Concurrent Process Model of Software Development", ACM SIGSOFT Vol. 18, No. 2, 1994*
6. *B. Henderson-Sellers, et al., The Object-Oriented Systems Life-Cycle, Comm. of the ACM, vol. 33, No.9, Sept. 1990, p145-159*
7. *W. Royce, "Managing the Development of Large Software Systems", IEEE WESCON, Aug. 1970, p1-9*
8. *R. Sharble£¬et al.£¬"The Object-Oriented Brewery: A Comparison of two Object-Oriented Development Methods"£¬ACM SIGSOFT Software Engineering Notes£¬Vol.18£¬No.2£¬1993*
9. *USA Department of Defense, Military Standard: Defense System Software Development, DOD-STD-2167A, Washington, Development. C., Feb. 1988*
10. *Jianxin Yan, "Expanding Rapid Prototyping Activity", Proceeding of IFIP CAPE'95:The 5th International Conference on Computer Application in Production and Engineering, Chapman & Hall, London, 1995*
11. *Jianxin Yan, On The Concurrent Engineering Approach to Software Development (Chinese Edition), Master Dissertation, Dept. of Computer Sci. & Eng., BUAA, 1996*
12. *Yang Wenlong, et al. ed., Software Engineering Additional Material, Dept. of Computer Sci. & Eng., Beijing Univ. of Aero. & Astro., 1991*

TOWARDS A VIRTUAL DESIGN ENVIRONMENT: A SHARED SEMANTIC PRODUCT MODEL

Jin-Kang Gui, Timo Ylä-Mononen and Harri Jokinen
VTT Manufacturing Technology
Technical Research Center of Finland (VTT)
Espoo, Finland

ABSTRACT

The rapid pace of advances in information technology have created a boom in research for a virtual engineering environment which supports collaborated design and rapid manufacturing. This paper attempts to make a special contribution to two of the key underlying technical issues: digital product and process modeling, and product information management. The concepts of semantic product modeling are uniquely developed as an evolution of the present 3D feature-based CAD technology. It is our belief that concurrent engineering in a distributed environment can be ultimately realized when the semantic product modeling has been thoroughly studied on an international scale.

KEYWORDS

virtual design environment (VDE), semantic product modeling (SPM), product modeling, product information modeling, concurrent engineering

1. INTRODUCTION

A number of pilot industrial projects have been recently reported as having successfully introduced a virtual product development (VPD) strategy into engineering design. For example, in 1995 Global Engineering Network (GEN) in Europe initiated aims to support intensively advanced engineering means, which would enable users and suppliers to get world-wide access to engineering knowledge(1). A "client" (a technical knowledge receiver) requests a "server" (a technical supplier) from a network of engineering solutions. In between, interaction or cooperation for design may be allowed to some extent. In this sense, GEN can be viewed as a networked "design by catalog".

To develop such a virtual design environment (VDE), technology faces many technical and social challenges: computing, communications, multimedia, standardization and social integration in an organization and in personnel where issues such as how a design team works effectively must be addressed. Typical tools available today to establish this environment include sophisticated solid modeling systems, various CAE systems, powerful computing networks with digital audiovisual supports and some standards like DXF, IGES and STEP. The achievements that have been demonstrated so far are mainly in supporting electronic commerce and development of some legacy products, i.e. types of "known" design , "Re-Used" or "Adapted" without conceptual changes in Workshop Design-Konstruktion (WDK) terminology(2). Because of a lack of matured technical solutions to some fundamental issues, the research is still at the infant stage.

The principal issues described in this paper are applicable to both networked and stand-alone applications for product design and product modeling. First, a general description of the concepts of a virtual design environment is given. Then the paper focuses on: a semantic product model to capture the semantics not only for the product but also for the development process, a modeling architecture called SPM-Shell which is compatible with hypermedia systems, and product information processing and management concerning the relations among CAD models, STEP models and PDM (Product Data Management) systems. Finally, further research is addressed.

2. A VIRTUAL DESIGN ENVIRONMENT

A CAD system includes a person, computer hardware, software and a certain type of problem(3). Similarly, a virtual design environment can be defined as (1) a virtual design team over various domains, (2) a virtual design office geographically composed of a collection of computing tools, and (3) virtual design activities whose main goal is to evaluate design alternatives.

Over the last decade, considerable research efforts have been made world-wide to investigate feature technology to support concurrent engineering[4]. As pointed out in a previous study[5], the main limitations to feature modeling are: (1) it is incapable of modeling a top-down design process; (2) mapping of viewpoint dependent features remains a myth[4], and therefore, (3) lack of a mechanism to facilitate communications and to accommodate various types of design knowledge in a people-centered and knowledge-intensive product development process. Figure 1 shows our idea of how feature technologies can be advanced toward a virtual design environment to support distributed concurrent engineering via semantic product modeling, communication technology, and AI methods.

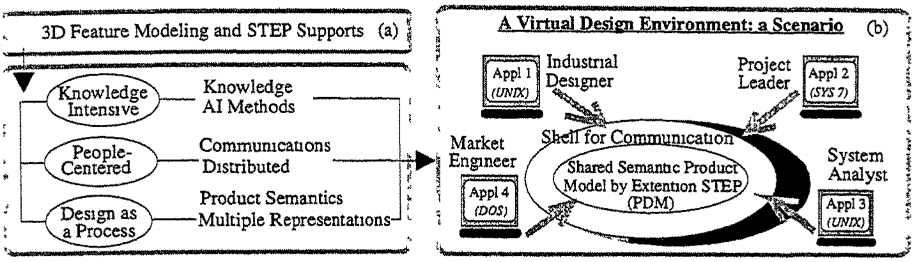


Fig. 1: From feature modeling to a virtual design environment

In general, an information infrastructure required for a VDE scenario shown in Figure 1 (b) consists of five basic elements: (1) an open semantic product modeling shell as part of application systems to capture a cooperative CAD process; (2) a shell for computer-supported cooperative work to control, to coordinate, to notify and to secure the team CAD process over the net; (3) product data management (PDM) to support life cycle by a shared semantic product model; (4) three kinds of standard supports for communications: semantically utilizing STEP for product information and knowledge exchanges and sharing; syntactically utilizing CORBA-like standards for software interoperability and TCP/IP and SGML standards for networking and information presentations; and (5) extensible and flexible integrated applications on a single end-front to support a talent designer playing multiple roles in a team.

3. SEMANTIC PRODUCT MODELING

The motivation behind semantic product modeling as an evolution of feature modeling is to model a product development process by using incomplete specifications of the product over real time or event-based design activities. In systematic design[6], a design process is distinguished by different phases of corresponding design tasks or activities. In WDK, the concept of "chromosome" is used to represent a "generic" product model for capturing a design history[7]. Recently, the STEP research community recognized a need for modeling and transferring "incorrect" STEP data[8]. Significant progress in the field of product modeling is attributed to early design supports through the development of assembly-based (or interchangeably called function-based) CAD systems[9]. A broad definition of semantic product modeling can be stated as "abstract machine descriptions and computing operations on them related to assembly design from brief to detail"[9]. It also addresses the product's life cycle.

3.1 SEMANTIC MODELING HYPERSPACE

In early research on solid modeling, some researchers used 4D space-time for dynamic interference analysis to test collisions between moving objects[10]. A similar step toward dynamic product modeling was recently made by introducing "time", function and behavior (physical processes) as the independent modeling dimensions[11][12]. A semantic product model defined in this hyperspace, in comparison with a feature model defined in feature spaces[13], is characterized as[11]:

- geometrically dimension-independent;
- multiple decomposition of the product to provide functional view and modular view;
- task-oriented or phase-related explanation of form feature semantics;
- capturing significant design features in the process which may not be a part of the real product;

- a one-to-many scheme for ease of communications, i.e. one master assembly model having multiple representations to various application models.

3.2 TIME-LABELED MULTI-GRAPH DATA MODEL

A multi-graph data structure as a computer model is used in our early realization to statically define an assembly model at the different levels of abstractions[5][14]. It is certainly useful for modeling the legacy products which are more or less known by their design process. Industrial practices indicate that:

- a design process of a prototype development for a new generation product is often unknown and may last many years. For instance, an innovative "bolted" car, SMART from Mercedes-Benz and Swatch from the "neat idea" stage to prototype lasted 25 years;
- design may be changed over time. That means the previous changes will be lost using a static model;
- design refinement from one stage to the next is often iteratively associated with a sequence of other design activities with other application models, and thus associated with a sequence of decision making. These types of design information, if recorded, must be beneficial for re-use.

It is obviously not easy to capture these development features by an inflexible static multi-graph. How a product can be dynamically defined in the hyperspace by a time-labeled multi-graph is illustrated in Figure 2. A product, Disk Joint, at each given time is represented by a collection of leaf nodes tinted in dark color and their relations in a multi-graph. A disk joint assembly is associated with this dynamic data model. New form features to be added each time, whenever necessary, are processed by four operations: feature inheritance, feature delegation, feature conversion and feature abstractions[11]. For example, the alignment datum features defined at T0 in Figure 2 is inherited by the next feature model defined at T1, in which the new feature, PatternHole must be co-axis to it.

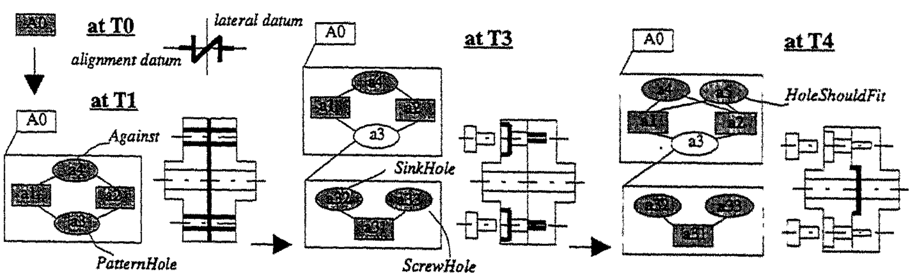


Fig. 2: Dynamically define Disk Joint by time-labeled multi-graph

3.3 SEMANTIC PRODUCT MODELING SHELL (SPM-SHELL)

The time-labeled multi-graph data model is built in a semantic product modeling shell (SPM-Shell) which resides on the top of CAD, or more generally, of distributed CAD in a VDE. The functionality of the shell is mainly to use this superstructure to navigate an assembly design or configuration process in a real design session, where a designer may record or associate his/her interests with related entities to the form features. Any decision making from the applications to contribute to the changes of forms either qualitatively or quantitatively is therefore easily captured and associated with this time-labeled multi-graph. To this end, the SPM-Shell will be realized as an environment to integrate various applications by adopting some concepts of DEXTER hypertext reference model[15].

The early DELTA system imitates a design team work with a top-down assembly and its computing architecture includes DesignPlanner, DesignConsultant and DesignSketcher[16]. The system realization bears a surprising similarity to a hypertext system by the DEXTER model which divides the system into three layers, the run-time layer, the storage layer and the within-component layer. By comparison, the MultiGraphBrowser in the DELTA Smalltalk environment corresponds to the run-time layer in DEXTER, DesignPlanner to the storage layer, and DesignConsultant and DesignSketcher to the within-component layer.

In DELTA, a shared multi-graph data model is used to link DesignPlanner to other two application objects. Similarly, the critical piece of the DEXTER model is a mechanism, called Anchor, as an interface to connect the hypertext network at the storage layer to the contents of particular components in the within-component layer. This mechanism maintains a separation between the storage and the applications so that it provides the possibility for end users to access, to view, and to manipulate the network structure dynamically through the run-time layer.

SPM-Shell as an advance development over the DELTA system is generally presented by a DEXTER-similar architecture as in Figure 3. The major concerns for this advanced development are (1) multimedia data types may be used for product representations; (2) node in the time-labeled multi-graph not only contains physical assembly geometric data but may also have other application contents such as a simulation object, and (3) isolated nodes at the storage layer are allowed to model unstructured information.

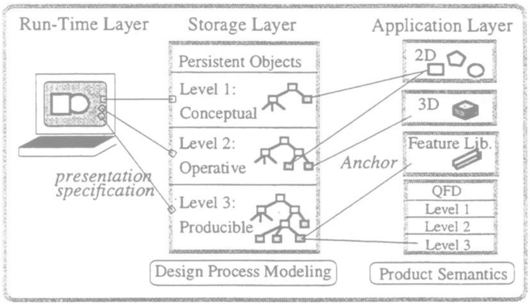


Fig. 3: A Dexter-like SPM-Shell Architecture

4. PRODUCT INFORMATION MANAGEMENT

Product information modeling by STEP is focused on information contents and meaning to be transferred by data, representations of the meaning, their exchanges and sharing[17]. This technology-independent model acts as a key for manufacturing industrial competitiveness and plays a central role in supporting communications in a virtual design environment. Following the idea to make a conceptual distinction between "information" (technology-independent) and "data" (technology-dependent)[18], we deal with three different but related models: a CAD model, a database data model and a STEP model and study how they can coherently act in a virtual design environment.

4.1 INFORMATION INTEGRATION

Figure 4 (a) illustrates two aspects of STEP integration: the ability to exchange digital data between the systems and shared distributed databases or a federated database system[19]. Two systems are integrated either through file exchange with a pre-processor on the sending side, a post-processor on the receiving side, or through archiving and accessing product data in a shared database. The latter integration is not as well-known as the former, which is particularly important for supports of a product's life cycle and requires a standard access interface, called SDAI in STEP.

Figure 4 (b) illustrates the latter integration: how product information is produced, stored, accessed and shared among applications.

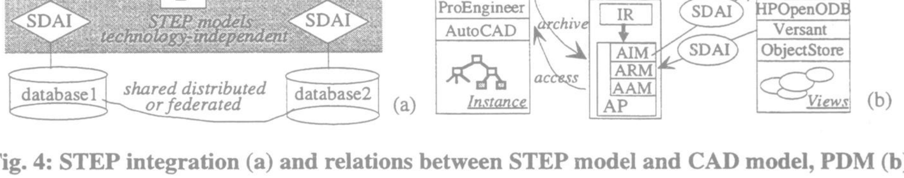


Fig. 4: STEP integration (a) and relations between STEP model and CAD model, PDM (b)

The conceptual schema in STEP is similar to but far more complicated than 3-schema architecture in a traditional database. STEP data generation involves data modeling by EXPRESS schemas. One major part of the STEP model is Integrated Resources (IR) ("context-free") used for different applications. The other is Application Protocols (AP) to define context and scope by Application Interpreted Model (AIM), and to specify information requirements for the designated application from Application Reference Model (ARM) developed and based on Application Activity Model (AAM) of the application domain.

During a design session, a CAD-related SPM-Shell will produce an instance of the product model. This instantiated product model, which supports the product life cycle, is achieved by a Product Data Management (PDM) system via STEP SDAI. No matter what kinds of CAD or PDM are used at the present or in the future, the information contents embedded in the CAD model and their meaning or views in the different contexts are specified by STEP standard models which have a longer lifetime than any particular technologies[18].

4.2 INFORMATION COMMUNICATION BY THREE-LEVEL SEMANTIC MODELING

A famous OSI (Open System Interconnection) Reference Model defines the framework for communication by splitting the communication tasks into seven layers[20]. Following this methodology, a three-level semantic product modeling is proposed to ease the design information communications in a VDE as shown in Figure 3 and 5. The product model for concurrent engineering represents not only a final producible product but also an integrated and generic administrative framework to capture the product development rational in team contexts. This implementation requires an extension of STEP models to have the capability to specify an incomplete product.

Communication within the same modeling level will follow the certain tasks specified at the corresponding level. Communication between adjacent levels involves refinement services from top to bottom to support creative design, and simplification services vice versus to support legacy product development. The services will be provided by STEP-based communication objects (i.e. Agents) which contain design rules and information in context.

The functionality for each level can be defined as follows. At a conceptual level, communication for design ideas is supported. The services at this level can involve (a) product or system partitioning; (b) function-physical principle-solution library; and (c) advanced material technology. At an operative level, task-based design activities will be communicated. The services can be (a) product design iterative sequence (regarding product information, who, when, where, what and how produces); (b) trade-off decision making sequence; (c) agent-based information or knowledge package within a particular domain; and (d) key engineering elements corresponding to customer requirements. At a producible level, production-based communication is supported. The services can be (a) manufacturing resources, (b) key production processes, (c) maintenance and so on.

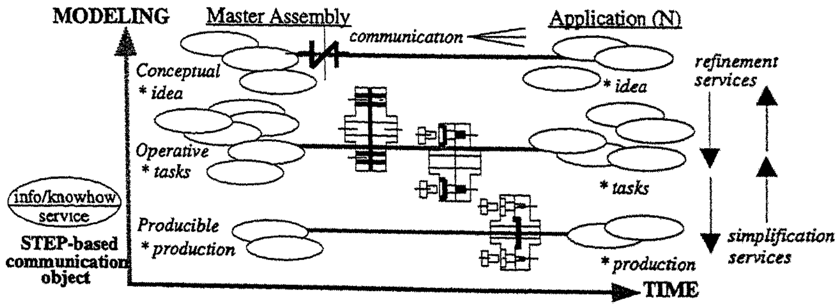


Fig. 5: Product information communication in space-time with three-level semantic modeling

5. FUTURE RESEARCH

This paper addressed two important technical issues for a VDE: product and process modeling, and product information management. To reach practical results, the following related issues must be further studied:

- different views to the product (various DfX);
- team work modeling (real social issues);
- knowledge processing (in STEP context, or de facto-standards); and
- 3D graphic user interface (beyond window standards).

6. ACKNOWLEDGMENT

The authors gratefully acknowledge the financial aid of VTT Manufacturing Technology. In particular, the work would not have been possible without the constant support and encouragement of Prof. Heikki Kleemola, a Research Director at VTT Manufacturing Technology. The ideas presented in this paper benefited from many fruitful discussions within our SPM-Shell consortium over the last year. We extend a special thanks to these members of the group: Dr. J. Rix and Mr. U. Jasnoch at FhG/IGD in Germany, Prof. M. M. Andreasen at DTU in Denmark, Prof. T. Kjellberg at KTH in Sweden, Mr. O. Myklebust at SINTEF in Norway and Prof. R. Joan-Arinyo at UPC in Spain.

7. REFERENCES

1. Rethfeld, U. and Thonemann , H.-G., "The Global Engineering Network GEN: Status and Political /Economical Impacts", Opening Productive Partnerships, K.-R. von Barisani et al. (Eds.), pp. 189-195, IOS Press, 1995
2. Hubka, V. and Eder, W. E., Theory of Technical Systems, Sprnger-Verlag, NY, 1988
3. Encarnacao, J and Schlechtendahl, E. G., Computer-Aıded Desıgn - Fundamentals and System Architectures, Springer-Verlag, 1983
4. Shah, J., "Features Technology: State-of-the Art Research Trends", Invited Lecture, FhG-IMA Workshop on Feature-Based Modeling, Genova, Italy, Nov. 1994
5. Gui, J.-K., "A Function-Behavior-Structure Machine Design Model and its Use in Assembly Sequence Planning", Journal of Engineerıng Design, Vol. 1, No. 3, pp. 239-259. 1990
6. Pahl, G. and Beitz, W., Engineering Design, edited by Wallace, K., The Design Council, (original German edition by Springer-Verlag 1977), 1984
7. Andreasen, M. M., "Modeling - The Language of the Designer", Journal of Engineering Design, Vol. 5, No. 2, pp. 103-115, 1994
8. Schlechtendahl, E. G., "ISO 10303 (STEP) Product Data Representation and Exchange: Implementation Methods", PDTAG-AM Tutorial on Implementing STEP ISO 10303, Darmstadt, Germany, Feb. 1995
9. Gui, J.-K., Methodology for Modeling Complete Product Assemblies, Acta Polytechnica Scandinavica (Mathematics and Computer Science Series 60), January 1993
10. Requicha, A. A. G. and Voelcker, H. B., "Solid Modeling: Current Status and Research Directions", IEEE CG & A, pp. 25-37, October 1983
11. Gui, J.-K., Ungerer, M. and Rix, J., Formulating Feature Links for Dynamic Product Information Modeling, a visiting report at FhG/IGD, Darmstadt, Germany, January 1995
12. Hale, M. A., Craig, J. I., Mistree, F. and Schrage, D. P., "DREAMS and IMAGE: A Model and Computer Implementation for Concurrent, Life-Cycle Design of Complex Systems", Concurrent Engineering: Research and Applications, Vol. 4, No. 2, pp. 171-186, June 1996
13. Shah, J., "Feature transformations between application-specific feature spaces", Computer-Aided Engineering Journal, pp. 247-255, December 1988
14. Gui, J.-K., "Object-oriented Assembly and Assembly Design Process Modeling", Journal of Engineering Design, Vol. 2, No. 2, pp. 141-149, 1991
15. Halasz, F. and Schwartz, M., "The DEXTER Hypertext Reference Model", Vol. 37, No. 2, ACM Communications (special issue), pp. 30-39, February 1994
16. Gui,'J.-K. and Mäntylä, M., "Functional understanding of assembly modeling", Computer-Aided Design, Vol. 26, No. 6 pp. 435-451, June 1994
17. Kjellberg, T. and Schmekel, H., "Product Modeling and 'Information-Integrated' Engineerıng Systems", Annals of the CIRP Vol. 41/1/1992, pp. 201-204
18. Wilson, P. R, "A View of STEP", in Wilson, P. R., Wozny, M. and Pratt, M. J. (Eds.), Geometric Modeling for Product Realization, North-Holland Netherlands, pp. 267-296, 1993
19. Yang, Y., "The Standard for the Exchange of Product Model Data (STEP): Architecture and Implementation", CALS Expo International '93, PS-4-25, September 1993
20. ISO 7498: Information processing systems - Open Systems Interconnection - Basic Reference Model, Beuth, Berlin 1988

SOLUTIONS TO FUNDAMENTAL PROBLEMS IN B-SPLINE CURVE DESIGN*

Kaihuai Qin[a,b], Youjiang Guan[c] and W Wang[a]
[a]Dept. of Computer Science, The University of Hong Kong, Hong Kong
[b]Dept. of Computer Sci. & Tech., [c]Dept of Precision Instrument & Mechanics,
Tsinghua University, Beijing, China

ABSTRACT

It has been found that there are identified problems with some existing fundamental algorithms in B-spline curve design. For instance, existing algorithms for degree raising fail to elevate the degree of uniform B-spline curves. In the paper, new methods for solving the problems are introduced. A few new algorithms for degree raising and knot insertion are presented. Practical examples of degree raising and knot insertion for B-spline curves are given. As a result of this paper, the conventional theory for degree raising of B-spline curves has to be revised.

KEY WORDS

B-spline, degree raising, curve, knot insertion.

1 INTRODUCTION

Knot insertion[1 2] and degree raising[2-6] of B-spline curves are two common methods in curve and surface design. Degree raising of B-splines plays a very important role in computer aided geometric design. It is frequently used to link curves of different degrees together to form either a composite curve or a sweeping surface. On the other hand, the knot insertion is widely used for modeling and displaying of B-spline curves and surfaces. The algorithms for knot insertion and degree raising of B-spline curves were studied in many published papers. Unfortunately, it has been found that the conventional algorithms have a lot of shortcomings, and sometimes incorrect results may be obtained with them. In fact, the conventional algorithms for degree raising of B-spline curves can be successfully used for only endpoint-interpolating B-spline curves and periodic ones. When used for degree raising of other B-spline curves, the conventional algorithms for degree raising will fail. When used for knot insertion of other B-spline curves except endpoint-interpolating curves and periodic ones, the conventional algorithms for inserting new knots into B-spline curves will fail to insert the knots near the ends of the knot vectors, or the boundary knots. Thus it is necessary to develop new methods for resolving such fundamental problems in B-spline curve design.

This paper is organized as follows: Section 2 states some problems in curve design. New methods for knot-insertion and degree raising of B-spline curves are introduced in the third section. New algorithms for improving existing knot-insertion algorithms and degree raising ones are presented in the fourth and the fifth sections, respectively. Finally, practical examples are given in the paper.

2 PROBLEMS OF THE TRADITIONAL ALGORITHMS FOR KNOT INSERTION AND DEGREE RAISING OF B-SPLINE CURVES

As mentioned above, there are flaws in existing algorithms for degree raising of B-spline curves, It is well known that Prautzsch's algorithm for degree raising[3,6] needs to insert the boundary knots into a curve in order to raise its degree. If the curve is a non-periodic uniform or generally nonuniform B-spline curve instead of an endpoint-interpolating curve, errors will occur in Prautzsch's algorithm for degree raising when one of the boundary knots is inserted into the curve by existing knot-insertion algorithms(e.g., Boehm's algorithm). We can find many such examples[7].

In order to resolve or avoid the above problems in knot-insertion and degree raising of B-spline curves, there may be the following ways:

- Development of new knot-insertion algorithms;
- Development of new degree-raising algorithms for B-spline curves;

* Project partly supported by the NSF of China.

- Improvement on existing knot-insertion algorithms[7];
- Improvement on existing degree-raising algorithms.

3 NEW METHODS FOR KNOT-INSERTION AND DEGREE RAISING OF B-SPLINE CURVES

Regardless of knot-insertion or degree raising of B-spline curves, mathematically it is the key point how to represent a set of B-splines of order k as linear combinations of other specific set of B-splines of order k or $k+1$. In this section formulae for representing a set of B-splines of order k as linear combinations of other specific set of B-splines of order k or $k+1$ are introduced, and an algorithm for degree raising of B-spline curves is given.

3.1 A New Method for Knot-Insertion

If a new knot $\hat{t}$ needs inserting into a curve, then one can use the following relationship[7] between the normalized B-splines defined over the knot vector

$$\mathbf{T}=\{t_0,\ t_1,\cdots,\ t_{k-1},\overbrace{t_k,\cdots t_k}^{m_1},\cdots,\overbrace{t_{k+s-1},\cdots t_{k+s-1}}^{m_s},\ t_{n+1},\cdots,\ t_{n+k}\}$$

and the normalized splines over the new knot vector $\hat{\mathbf{T}}$ obtained by inserting the knot $\hat{t}$ into the knot vector $\mathbf{T}$:

$$N_{j,k}(t)=\begin{cases}\hat{N}_{j+1,k}(t), & \hat{t}\le t_j\\ \dfrac{\hat{t}-t_j}{t_{j+k-1}-t_j}\hat{N}_{j,k}(t)+\hat{N}_{j+1,k}(t), & t_j\le\hat{t}\le t_{j+1}\\ \dfrac{\hat{t}-t_j}{t_{j+k-1}-t_j}\hat{N}_{j,k}(t)+\dfrac{t_{j+k}-\hat{t}}{t_{j+k}-t_{j+1}}\hat{N}_{j+1,k}(t), & t_{j+1}\le\hat{t}\le t_{j+k-1}\\ \hat{N}_{j,k}(t)+\dfrac{t_{j+k}-\hat{t}}{t_{j+k}-t_{j+1}}\hat{N}_{j+1,k}(t), & t_{j+k-1}\le\hat{t}\le t_{j+k}\\ \hat{N}_{j,k}(t), & t_{j+k}\le\hat{t}\end{cases}\tag{1}$$

Eq.(1) can be described by a unified form as follows:

$$N_j^k(t)=\alpha_j\hat{N}_j^k(t)+\beta_j\hat{N}_{j+1}^k(t)$$

If $t_\ell<\hat{t}\le t_{\ell+1}$, using Eq.(1) one can obtain new control vertices:

$$\begin{cases}\hat{\mathbf{V}}_j=\mathbf{V}_j,\quad j=k-1,k,\cdots,n\\ \begin{bmatrix}\hat{\mathbf{V}}_0\\ \hat{\mathbf{V}}_1\\ \vdots\\ \hat{\mathbf{V}}_{k-2}\end{bmatrix}=\begin{bmatrix}\beta_0 & \alpha_1 & & & 0\\ & \beta_1 & \alpha_2 & & \\ & & \ddots & \ddots & \\ 0 & & & \beta_{k-2} & \alpha_{k-1}\end{bmatrix}\begin{bmatrix}\mathbf{V}_0\\ \mathbf{V}_1\\ \vdots\\ \mathbf{V}_{k-1}\end{bmatrix}\\ \hat{\mathbf{T}}=\{t_1,\ t_2,\ \cdots,\ t_{n+k}\}\cup\{\hat{t}\}\end{cases}\tag{2}$$

if $0\le\ell<k-1$, or $\hat{t}\le t_{k-1}$;

$$\begin{cases}\hat{\mathbf{V}}_j=\mathbf{V}_j, & j=0,1,\cdots,\ell-k+1\\ \hat{\mathbf{V}}_j=(1-\alpha_j)\mathbf{V}_{j-1}+\alpha_j\mathbf{V}_j, & j=\ell-k+2,\cdots,\ell\\ \hat{\mathbf{V}}_j=\mathbf{V}_{j-1}, & j=\ell+1,\cdots,n+k\\ \hat{\mathbf{T}}=\{t_0,\ t_1,\ \cdots,\ t_\ell,\ \hat{t},\ t_{\ell+1},\ \cdots,\ t_{n+k}\}\end{cases}\tag{3}$$

if $k-1\le\ell\le n$ and $\hat{t}\ne t_{n+1}$, or $t_{k-1}<\hat{t}<t_{n+1}$, where $\alpha_j=(\hat{t}-t_j)/(t_{j+k-1}-t_j)$;

$$\begin{cases} \hat{\mathbf{V}}_j = \mathbf{V}_j, \quad j = 0,1,\cdots,n-k+1 \\ \begin{bmatrix} \hat{\mathbf{V}}_{n-k+2} \\ \hat{\mathbf{V}}_{n-k+3} \\ \vdots \\ \hat{\mathbf{V}}_n \end{bmatrix} = \begin{bmatrix} \beta_{n-k+1} & \alpha_{n-k+2} & & & 0 \\ & \beta_{n-k+2} & \alpha_{n-k+3} & & \\ & & \ddots & \ddots & \\ 0 & & & \beta_{n-1} & \alpha_n \end{bmatrix} \begin{bmatrix} \mathbf{V}_{n-k+1} \\ \mathbf{V}_{n-k+2} \\ \vdots \\ \mathbf{V}_n \end{bmatrix} \\ \hat{\mathbf{T}} = \{t_0, \ t_1, \ \cdots, \ t_{n+k-1}\} \cup \{\hat{t}\} \end{cases} \tag{4}$$

if $\ell \geq n$, or $\hat{t} \geq t_{n+1}$. Eqs.(3)-(4)can be used implementation of a general knot-insertion algorithm.

3.2 A new algorithm for degree raising

It can be proved that there is an identity representing B-splines of order k as linear combinations of B-splines of order $k+1$as follows[8]:

$$\begin{bmatrix} B_{i-k+1,k}(t) \\ \vdots \\ B_{i-1,k}(t) \\ B_{i,k}(t) \end{bmatrix} = \frac{1}{k} \begin{bmatrix} a_{0,0}(i) & a_{0,1}(i) & \cdots & a_{0,k}(i) \\ a_{1,0}(i) & a_{1,1}(i) & \cdots & a_{1,k}(i) \\ \vdots & \vdots & \cdots & \vdots \\ a_{k-1,0}(i) & a_{k-1,1}(i) & \cdots & a_{k-1,k}(i) \end{bmatrix} \begin{bmatrix} B_{i-k,k+1}(t) \\ \vdots \\ B_{i-1,k+1}(t) \\ B_{i,k+1}(t) \end{bmatrix}, t \in [t_i, \ t_{i+1}] \tag{5}$$

where

$$a_{u,v}(i) = \begin{cases} (-1)^{u-v} \sum_{s=1}^{k-u} \dfrac{\alpha_{u,v}(i)}{1-\gamma_{s-1}(i-k+u+1)}, & u \geq v; \\ (-1)^{v-u+1} \sum_{s=k-u-1}^{k-1} \dfrac{\beta_{u,v}(i)}{\gamma_{s+1}(i-k+u+1)}, & u+1 \leq v; \end{cases} \quad u = 0,1,\cdots,k-1; v = 0,1,\cdots,k.$$

$$\alpha_{u,v}(i) = \begin{cases} 1, & u = v; \\ \prod_{j=0,}^{u-v-1} \dfrac{\gamma_{s+j}(i-k+u-j)}{1-\gamma_{s+j}(i-k+u-j)}, & u > v; \end{cases}$$

$$\beta_{u,v}(i) = \begin{cases} 1, & u+1 = v; \\ \prod_{j=0,}^{v-u-2} \dfrac{1-\gamma_{s-j}(i-k+u+j+2)}{\gamma_{s-j}(i-k+u+j+2)}, & u+1 < v; \end{cases}$$

$$\gamma_s(j) = \frac{t_{j+s} - t_j}{t_{j+k} - t_j}, \quad s = 0,1,\cdots,k.$$

If a B-spline curve defined over the knot vector $\mathbf{T}$ needs degree raising, then the knot vector should be refined as follows

$$\hat{\mathbf{T}} = \{t_{-1}, t_0, \ t_1, \cdots, \ t_{k-1}, \overbrace{t_k, \cdots t_k}^{m_1+1}, \cdots, \overbrace{t_{k+s-1}, \cdots t_{k+s-1}}^{m_s+1}, \ t_{n+1}, \cdots, \ t_{n+k}, t_{n+k+1}\}.$$

By means of Eq.(5), it is easy to obtain a formula for Computation of the control vertices of the ($i-k+1$)st segment of curve $\mathbf{C}^{i-k+1}(t) = \mathbf{V}^{i^T}\mathbf{N}_i^k(t) = \hat{\mathbf{V}}^{i^T}\hat{\mathbf{N}}_i^{k+1}(t)$ ($t \in [t_i, \ t_{i+1}]$; $i = k-1, k, \cdots, n-k+1$)

$$\left[\hat{\mathbf{V}}^i\right]^T = \frac{1}{k}\left[\mathbf{V}^i\right]^T \begin{bmatrix} a_{0,0}(i) & a_{0,1}(i) & \cdots a_{0,k}(i) \\ a_{1,0}(i) & a_{1,1}(i) & \cdots a_{1,k}(i) \\ \vdots & \vdots & \cdots \vdots \\ a_{k-1,0}(i) & a_{k-1,1}(i) & \cdots a_{k-1,k}(i) \end{bmatrix}$$

where $\mathbf{V}^i = [\mathbf{V}_{i-k+1}, \mathbf{V}_{i-k+2}, \cdots, \mathbf{V}_i]^T$ and $\mathbf{N}_i^k(t)$ is a row vector composed of $B_{j,k}(t)$ ($j=i-k+1, \cdots, i$), and its algorithm:

> **Algorithm 1**
>
> *Insert s knots into the curve by a general knot-insertion algorithm*[7] *to get* $\hat{\mathbf{T}}$;

```
j = 0, J = 0;
for (i=k-1 to n+s by m_J+1)
    if (i == k-1) Start = 0;
    else  Start  =  k - m_J  + 1;
    for (v=Start to k by 1)
        Ṽ_j = 0;
        for (u=0 to k-1 by 1)  Ṽ_j = Ṽ_j + a_{u,v}(i) V_{i-k+1+u};
        j = j + 1;
    endfor
    J = J+1;
endfor
```

4 IMPROVEMENT OF EXISTING KNOT-INSERTION ALGORITHMS

As mentioned in Section 1, Boehm's algorithm can be used only for endpoint-interpolating B-spline curves and periodic ones. If we want to insert $\hat{t}$ ($t_0 \le \hat{t} < t_{k-1}$ or $t_n < \hat{t} \le t_{k+k}$) into a curve with finite control vertices except endpoint-interpolating B-spline curves and periodic ones, an improvement on Boehm's algorithm, i.e., Algorithm 2, is another alternative except use of Eqs.(2)-(4). Its basic idea is as follows:

- First, extend the curve to pass the first and the last control vertices, respectively, by means of elevating the multiplicity of the first and the last knots as well as that of the first and the last control vertices to k, respectively.
- Second, insert the knot $\hat{t}$ into the curve by Boehm's algorithm for knot insertion.
- Finally, remove the surplus knots and control vertices so that the domain of the resulting curve remains unchanged.

```
Algorithm 2
EndPoint_Interpolate_Flag = TRUE;
for (i=0 to k-2 by 1)
    if (t_i ≠ t_{i+1})
        EndPoint_Interpolate_Flag = FALSE;
        break;
    endif
endfor
if (EndPoint_Interpolate_Flag = TRUE)
    for (I = n to n+k-2 by 1)
        if (t_i ≠ t_{i+1})
            EndPoint_Interpolate_Flag = FALSE;
            break;
        endif
    endfor
endif
if (EndPoint_Interpolate_Flag = TRUE) insert t̂ into the curve by Boehm's Algorithm;
else
    elevate the multiplicity of the first and the last knots to k, and elevate the multiplicity of the
    first and the last control vertices to k in order to convert the curve into an endpoint-
    interpolating curve by extending the curve;
    insert t̂ into the curve by Boehm's Algorithm;
    remove the surplus knots and control vertices so that the domain of the curve remains
    unchanged;
endelse
```

When substituting Boehm's algorithm in Algorithm 2 by Oslo algorithm for knot-insertion, we can improve Oslo algorithm such that it can be used for knot-insertion of various B-spline curves regardless of endpoint-interpolating B-spline curves or other ones.

5 IMPROVEMENT OF THE TRADITIONAL ALGORITHMS FOR DEGREE RAISING OF B-SPLINE CURVES

In this section, we will present three ways to improve the traditional algorithms for degree raising of B-spline curves.

I. Curve extension-based algorithm

Algorithm 3

i) Add *(k-1)* knots and *(k-1)* control vertices at the start point and end point of the curve, respectively:

$$\overbrace{t_0,\cdots,t_0}^{k-1};\ \overbrace{\mathbf{V}_0,\cdots,\mathbf{V}_0}^{k-1};\quad and\ \overbrace{t_{n+k},\cdots,t_{n+k}}^{k-1};\ \overbrace{\mathbf{V}_n,\cdots,\mathbf{V}_n}^{k-1}.$$

ii) Elevate the degree of the curve using the conventional algorithms for degree raising of B-spline curves[3-6]. The new control vertices and knots are indicated by

$$\hat{t}_0,\cdots\hat{t}_{\hat{n}+k+1}\ and\ \hat{\mathbf{V}}_0,\cdots,\hat{\mathbf{V}}_{\hat{n}}.$$

The number of the new control vertices is equal to $\hat{n}+1$.

iii) Remove the added curve segments and the associated knots as follows:

$$\hat{t}_0,\cdots\hat{t}_{2k-3};\ \hat{\mathbf{V}}_0,\cdots,\hat{\mathbf{V}}_{2k-3};\quad and\ \hat{t}_{\hat{n}-k+4},\cdots\hat{t}_{\hat{n}+k+1};\ \hat{\mathbf{V}}_{\hat{n}-2k+3},\cdots,\hat{\mathbf{V}}_{\hat{n}}.$$

II. Endpoint-interpolating-based algorithm

Algorithm 4

i) for (j=*1* to *k-2* by *1*)

Insert knots $\hat{t} = t_{k-1}$ and $\hat{t} = t_{n-1}$ using Eq. (1);

ii) Remove *(k-2)* redundant control vertices $\hat{\mathbf{V}}_0,\hat{\mathbf{V}}_1,\cdots,\hat{\mathbf{V}}_{k-3}$, *and* $\hat{\mathbf{V}}_{n+k-1},\hat{\mathbf{V}}_{n+k},\cdots,\ \hat{\mathbf{V}}_{n+2(k-2)}$ at the ends of the curve.

iii) Let $t_{k-2} = t_{k-3}$ and $t_{n+2k-2} = t_{n+2k-3}$.

iv) Remove the surplus knots at the ends of the knot vector: $t_0, t_1, \cdots, t_{k-3}$, and $t_{n+2k-1}, t_{n+2k}, \cdots, t_{n+3k-4}$.

v) The degree of the curve is elevated by means of the conventional algorithms for degree raising of B-spline curves[3-6].

III. Improvement of Prautzsch's algorithm for degree raising based on a general knot-insertion method

As mentioned in Section 1, the main cause for Prautzsch's algorithm for degree raising to fail is that the traditional algorithms for knot-insertion fail to insert boundary knots into some curves such as uniform B-spline curves. It is a natural idea to substitute the traditional knot-insertion algorithm by a general one. Fortunately, such a general method for knot insertion has been found[7]. Therefore, it is an intuitive improvement on Prautzsch's algorithm for degree raising of B-spline curves that Boehm's algorithm is substituted by the general knot insertion one in Prautzsch's algorithm. The detailed algorithm is omitted here.

6 EXAMPLES

Given a uniform B-spline curve of order *4* defined over the knot vector *{0,1,2, 3,4,5,6,78,9,10}*. Let $\hat{t}$ =*2*, then the new control vertices got by Eq.(2) are shown in Fig. 1.

As shown in Figures 2-4, the degree of a given uniform B-spline curve is elevated from degree *3* to *4* by Algorithms 2-4, respectively. In part (a) of each figure the original control polygon of the uniform B-spline curve of degree *3* is drawn. The original knot vector of the curve is as follows:

{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11,12, 13, 14}.

Figure 2 (b)-(d) illustrate the degree-raising process of Algorithm 2. In Figure 2 (b), *(k-1)* knots and *(k-1)* control vertices at the start point and the end point of the curve

$0, 0, 0, V_0, V_0, V_0$ and $14, 14, 14, V_7, V_7, V_7$

are added at its ends to extend the curve to V_0 and V_7, respectively. Then, elevate the degree of the extended curve as shown in Figure 2 (c). Finally, remove the added curve segments and the associated control vertices to get the degree-raised control polygon of the given curve as indicated in Figure 1 (d). The *4*th degree B-splines are defined over the knot vector as follows:

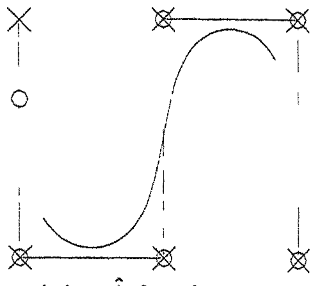


Fig.1 Insert the knot $\hat{t}$ =2 into the curve
× initial control vertices, O new control vertices

{1, 2, 2, 3, 3, 4, 4, 5, 5, 6, 6, 7, 7, 8, 8, 9, 9, 10, 10, 11, 11, 12, 12, 13}. (6)

Figure 3 (b)-(d) illustrate the degree-raising process of Algorithm 3. After the knots *3* and *11* are inserted twice, the new control polygon is obtained as shown in Figure 3 (b). Figure 3 (c) shows the result obtained by removing the redundant control vertices and the surplus knots at the ends of the curve from Figure 3 (b). The degree-raised control polygon of the curve is shown in Figure 3 (d). The *4*th degree B-splines are defined over the following knots:

{3, 3, 3, 3, 3, 4, 4, 5, 5, 6, 6, 7, 7, 8, 8, 9, 9, 10, 10, 11, 11, 11, 11, 11}. (7)

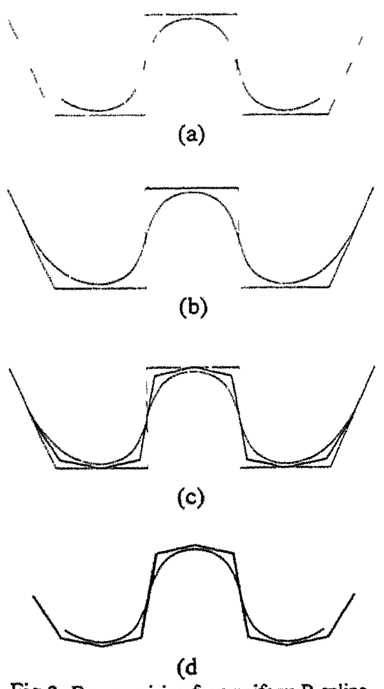


Fig.2 Degree raising for a uniform B-spline curve of degree 3 by **Algorithm 2.**

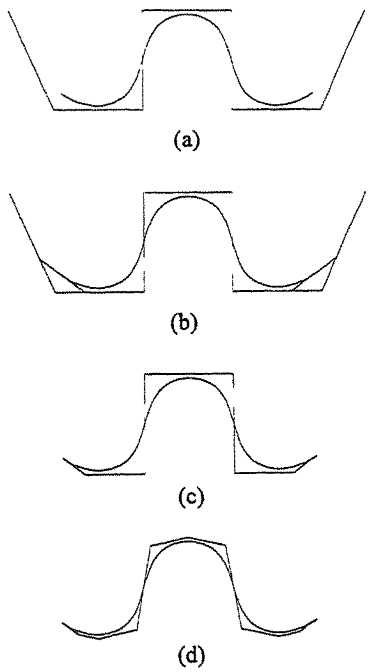


Fig.3 Degree raising for a uniform B-spline curve of degree 3 by **Algorithm 3.**

Figure 4 (b) shows the degree-raised control polygon obtained by Algorithm 1. The curve is the same curve as Figs. 2 and 3. The knot vector for degree raising of the given curve from degree 3 to 4 is chosen as follows:

{-1, 0, 1, 2, 3, 4, 4, 5, 5, 6, 6, 7, 7, 8, 8, 9, 9, 10, 10, 11, 12, 13, 14, 15}. (8)

If inserting the knots into the curve such that Equation (8) is substituted by Equations (6) and (7) we can obtain Fig. 2 (d) and Fig. 3 (d) by Algorithm 1, respectively. It is shown that Algorithm 1 is more general than the other algorithms for degree raising.

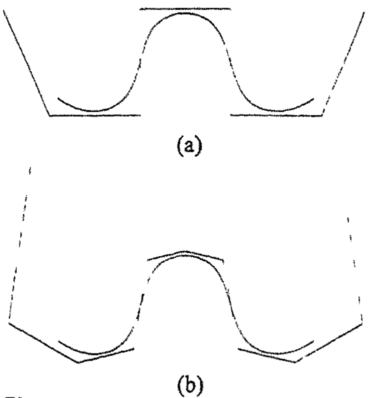


Fig.4 Degree raising for a uniform B-spline curve of degree 3 by **Algorithm 1.**

REFERENCES

1. Boehm, W., Inserting new knots into B-spline curves, *CAD*, 12:4(1980), p.199-201.
2. Cohen, E., Lyche, T., & Riesenfeld, R, Discrete B-splines and subdivision techniques in computer-aided geometric design and computer graphics, *Computer Graphics and Image Processing*, 14(1980), p.97-111.
3. Prautzsch, H., Degree elevation of B-spline curves, *CAGD*, 1(1984), p.193-198.
4. Cohen, E., Lyche, T.& Schumaker, L. L., Algorithms for degree-raising of splines, *ACM Transactions of Graphics*, 4:3(1985), p.171-181.
5. Cohen, E., Lyche, T.& Schumaker, L. L., Degree raising for splines, *Journal of Approximation Theory*, 46(1986), p.170-181.
6. Prautzsch, H. & Piper, B., A fast algorithm to raise the degree of spline curves, *CAGD*, 8(1991), p.253-265.
7. Qin, K., A general method for inserting knots into B-spline curves and its applications, submitted to CAD, revised.
8. Qin, K., A matrix method for degree raising of B-spline curves, Science in China(Series E), 27:1(1997), p.71-81.

Neural network method to reconstruct the freeform surfaces

Wang kai   Zhang caiming

(Computer Science Department, Shandong university, Jinan, Shandong, 250100)

Abstract: This article presents a artificial neural network approach to solve the problem of reconstruction and manufacturing of freeform surfaces in reverse engineering. Take advantage of the global minimum property of Simulated Annealing Procedure, a technique is proposed to accept a temporally failed training result in accordance to probability, with this technique, the training can jump out of the local minimum and converge to the global minimum. The method was tested to be better than the algorithm given in article [9], when used to solve the problem of reconstruction and manufacturing of freeform surfaces.

Keywords: Reverse Engineering   Freeform Surfaces   Neural Networks′   BP Algorithm   Simulated Annealing

INTRODUCTION

Freeform surfaces is a kind of typical industrial curve surfaces, for example, the outlook of a plane or a car, which is characterized by the smooth curvature. In CAD/CAM, the reconstruction and manufacturing of freeform surfaces have been important research fields for a long time [1-8], and have wide applying background in the factories such as automobile, aircraft and shipbuilding. In these factories, there exist various of reverse engineering problems, one of the most common problem is how to create the mathematics equations of an existing object, for example, in the aircraft industries, the researchers design a physical model based on the functional requirements and analysis, in order to study or produce it, they must know the mathematics representations of this model, that is concerned with a problem of reconstruction of freeform surfaces.

In article [9], P GU and X YAN proposed a neural network trained by BP algorithm to solve the problem of reconstruction of freeform surfaces, as the BP algorithm is not a linear method, there must exist the local minimum, furthermore, it is difficult to choose appropriate parameters for the network, therefore, the algorithm can hardly reach the expected requirement, and sometimes may create terrible errors. On account of these drawbacks, this article gives a new algorithm which connects the BP algorithm with the Simulated annealing algorithm[10], it preserves the error's back propagation method of BP algorithm, and uses the gradient descent method to adjust the weights of neural networks, with respect to the training result, the new algorithm accepts the successful result, but it does not reject the failed result and accept it by probability. Compared with the single BP algorithm of article[9], the new algorithm has not only the advantage of quick descent of BP algorithm, but also the global minimum, better robust and more freedom to choose the parameter.

MODEL OF THE NEURAL NETWORK

For the propose of the reconstruction of an existing freeform surfaces, first the surface was digitized to get some scattered points by a laser scanner or other method[11, 12], then these points are used as sampling points to train the neural network.

Figure 1 shows the topological structure of the neural network. this is a forward network without feedback. including input layer, output layer and several hidden layers, there are no links among the nodes of the same layer, the information is transformed layer by layer, here x_i is the input of the network, y_j is the output of the network..

It is well known that the training and learning of a neural network are just to change the weights among the nodes, in this article, the weights are adjusted according to the errors between the real output and the expect output during the training procedure.

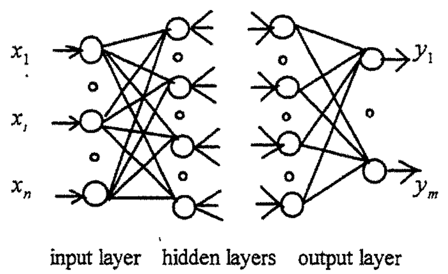


Figure 1 The topological structure of neural networks

The algorithm procedure

The Simulated Annealing algorithm introduced in article [10] and BP algorithm are used to train the designed network. The Simulated Annealing algorithm is a heuristic random search algorithm which came from the heat balance problem in physics, the most useful property of it is the global minimum property. In the crystal physics, to make the material reach the lowest energy state, the material should be heated to melt, thus it can be in a high energy state, then the temperature of the material is decreased step by step to make it coagulate, if the temperature drops slowly enough near the coagulating point, the material is sure to form the lowest energy state. There is something in common with respect to the procedure of optimal problem, during the procedure, each point in the solution space of the optimal problem can be seen as a solution, each solution has its own object function, the purpose of optimize is just to find the minimum or maximum object functions in the solution space.

Once a neural network structure is designed, the weights W of it can be seen as the micro state of material system and the error between the real outputs and expect outputs can be seen as the inner energy, furthermore, the purpose of training the network is to find the appropriate W to make the inner energy e minimal, thus the Simulated Annealing procedure can be used to train the network. The procedure id described briefly as follows: first set a parameter T to analog the annealing temperature, then compute the difference Δe between the present inner energy e and the former inner energy e under the temperature T, accept the training weights in accordance to the probability $\exp(-\Delta e / T)$,decrease the temperature, repeat the procedure for many times, only if T drops slowly enough and $T \to 0$, the network can stop at the state of minimal e.

Next, we introduce how to train the network by our algorithm, the algorithm is given in details below.

(1) Initialize the temperature T, step η, momentum α, system control threshold p^*, q^*, set iterate time $m = 0$, to node j, set threshold θ_i and weight $w_{i,j}$ of node i to j to be small random numbers between 0 and 1, at the same time, set the weights to be the optimize weights at present, set the system minimal error and present minimal error to be $Min\_error = Max$, where Max is a big enough number.

(2) Set $k = 0$, $m = m + 1$.

(3) Set $k = k + 1$, to input pattern $X(k) = \{ x_i(k) \}_i$,

the expect output pattern is $Z(k) = \{ z_j(k) \}_j$,

use following equations to compute the real output $Y(k) = \{ y_j(k) \}_j$, $k = 1......N$.

$$\begin{cases} v_j(k) = \sum_i w_{i,j}(k) y_i(k) - \theta_j(k) \\ y_j(k) = \dfrac{1}{1 + e^{-v_j(k)}} \end{cases}$$

(4) For output $Y(k)$ of each time, calculate the error,

$$e^m{}_j(k) = 1/2(z_j(k) - y_j(k))^2, j \in Out\_node \quad ,$$

where Out_node is the collection of output nodes.

If $k = N$, then all the sample points have been used to train the network, go to step (5); else, go to step (6).

(5) If $\Delta e = \sum_{k=1}^{N} \sum_{j \in Out\_node} e_j^m(k) - \sum_{k=1}^{N} \sum_{j \in Out\_node} e_j^{m-1}(k) < 0$,

the training make the error less, it succeeds, accept the weights of this training and increase the step length: $\eta = \eta * a, a > 1$,

if $\sum_{k=1}^{N} \sum_{j \in Out\_node} e_j^m(k) < Error$,

then set the control threshold $p = 0$, set the present error: $Error = \sum_{k=1}^{N} \sum_{j \in Out\_node} e_j^m(k)$,

set the present weights to be the optimal weights W^*, go to step (6);

else, let $p = p + 1$, go to step (6);

if $\Delta e = \sum_{k=1}^{N} \sum_{j \in Out\_node} e_j^m(k) - \sum_{k=1}^{N} \sum_{j \in Out\_node} e_j^{m-1}(k) \geq 0$,

then $p = p + 1$, accept the weights according to probability $\exp(-\Delta e / T)$, if the weights

are accepted, then increase the step length: $\eta = \eta * a, a > 1$, go to step (6);
else, decrease the step length: $\eta = \eta * b, b < 1$, set the momentum α to be 0, go to step(2).

(6) If $p > p^*$, it illustrates that it is difficult to find a smaller error than $Error$ under temperature T, go to step (8);
else go to step (7).

(7) Use the next two equations to adjust the weights,

1) Back propagate the error information:

$$\sigma^m{}_j(k) = \begin{cases} e_j^m(k) y_j(k)[1 - y_j(k)], & j \in Out\_node \\ y_j(k)[1 - y_j(k)] \sum_l \sigma_l^m(k) w_{l,j}(k), & j \notin Out\_node \end{cases}$$

2) Decrease the weights according to the gradient direction:

$$w_{i,j}^m = w_{i,j}^m + \alpha[w_{i,j}^m - w_{i,j}^{m-1}] + \eta \sum_k \sigma_j^m(k) * y_i(k)$$

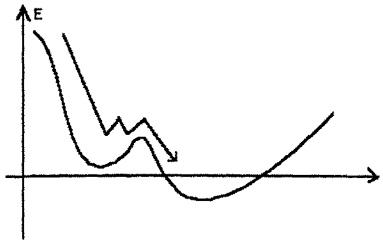
go to step (2).

(8) Decrease the temperature : $T = T * c, c < 1$,
now, if $Error < Min\_error$,
then set the system minimal error $Min\_error = Error$, the system optimal weights $W^{**} = W^*$, let $q = 0$;
else, set $q = q + 1, Error = Min\_error$.

(9) If $q > q^*$, it indicates that the temperature has been dropped for many times but the error does not decrease, so there is no necessary to reduce the temperature, output the system optimal weights, stop;
else, set the present optimize weights $W^* = W^{**}$ and the present system minimum error $Error = Min\_error$ go to step (2).

DISCUSSIONS AND EXAMPLES

From the 5th step of the algorithm we can see that when it meets the successful result, the algorithm is the same as the BP algorithm in article [9], however, with respect to the failed result, the algorithm does not abandon it as article [9] does, but accept it according to probability, which makes the algorithm can jump out of the local minimum, because according to the viewpoint of energy function (figure 3 and 4 show the energy descent manners of the two algorithms), there are several local minimums during the procedure of energy descent to global minimum, sometimes the energy may increase, but this increase is only temporary and will make the energy decrease to global minimum at last, so in this article, the algorithm admits the search of the solution go along the direction that may make the solution worse and the energy increase to avoid the local minimum, only if the annealing temperature is high enough and the training time is long enough, the algorithm can reach the global minimum of the energy, For other words, the global optimal solution [10].



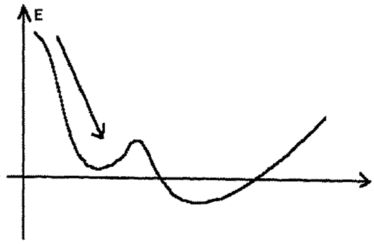


Figure 2 The energy descent manner of the algorithm of our article figure 3 The energy descent manner of BP algorithm in article [9]

Furthermore, we can see from the set of error *Error*, *Min_error* that when the algorithm stops, the output values must to be the best solution during the procedure of the solution search and the threshold p^*, q^* can guarantee the efficiency of the algorithm, thus make it possible to reach the optima result, though the algorithm do not choose the most appropriate parameters.

In order to examine the efficiency of our algorithm, we compare it with the algorithm in article [9], the two algorithm adopt the same neural network structures used in article [9] as showed in figure 4, this is a 4-layer neural network, which is composed of one output node, two input node and two hidden layers each including 10 nodes. During the training procedure, the input u , v of neural network are the parameters of NURB curve surface, the output is the space coordinates of reconstructed curve surface. The neural network is trained by the NURB data provided by article [9], the other point values on the NURB curve surface are used to examine the effect of the network, the comparison results are showed in table 1, for convenient, the table only gives the value of x coordinate, where x1 is the expect value taken from NURB surface, x2 is the result of the algorithm in article [9], x3 is the result of our algorithm through 4000 circulation(the beginning step length is 1.4, the momentum is 0.4, the temperature is 1.0).

From table 1, we can see that the average error of our algorithm is about 0.01, much closer to the real value than that of the algorithm in article [9], which is about 0.02, so the neural network created using our algorithm is better than that of article [9]'s, it can reflect the real condition of the NURB curve surface better.

The relationship between the training time and error in algorithms is showed in figure 5, where the thin line shows the new algorithm, the thick line shows that of the algorithm in article[9], the horizontal axle is the time and the vertical axle is the error *Min_error*. The figure demonstrates that at the beginning, both of the algorithms'errors undergo a period of slow descent, then begin to quick descent almost at the same time, but the algorithm of article [9] can not drop more when it reaches some level, on the contrary, the new algorithm's error can decrease continually with the time increasing, except the descent speed changes to be slow.

Several different step lengths and momentum values have been set to the new algorithm in this example, and gotten results which make the errors small. But with respect to the algorithm of article [9], if we do not choose the parameters in algorithm of article [9] carefully, the algorithm result may seem terrible. This demonstrates that the new algorithm have more freedom in choosing the parameters than the algorithm in article [9].

Apply the new approach to the reconstruction of other curve surfaces such as cubic Bezier surfaces, we get satisfied results too.

CONCLUSIONS

The neural network approach to reconstruct the freeform surfaces has great meaning in CAD/CAM, it simulates the inner relationship among points on the curve surfaces by training and learning of neural network nodes using sampling points, thus we can know the coordinates of the other points on the surfaces without computing the specific mathematics equation, when only given a limited number of points information.

In this article, the new algorithm take the advantages of the Simulated Annealing algorithm and the BP algorithm to improve the neural network approach solving the reconstruction problem of freeform surface and decrease the difficult level of parameter choosing, but there is still a weakness in it, that is the slow training speed, in the example of this article, the iterate number is about 4000 to x coordinate, furthermore, when the curvature of freeform surface get larger or the structure get more complex, the training speed becomes slower.

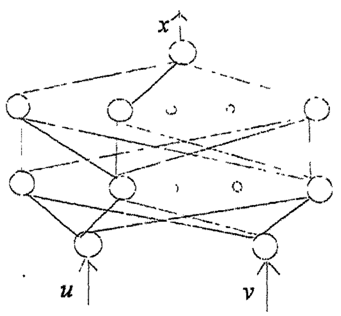


Figure 4 The structure of neural network to reconstruct the NURB curve surface

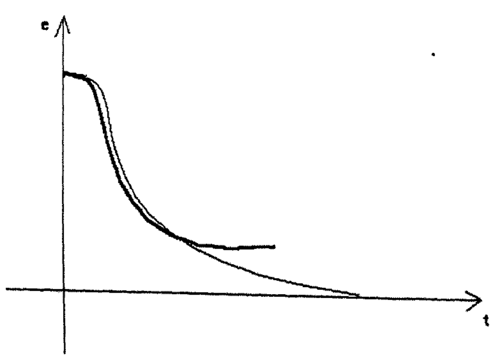


Figure 5 The relationship between error and time

u	v	x1	x2	x3
1.000000	1.000000	11 500000	11.447000	11.487403
1.000000	1.100000	12.264167	12.275300	12.266978
1 000000	1.200000	13 053333	13.102600	13.042305
1.000000	1 300000	13 862500	13.928400	13.870028
1 000000	1 400000	14 686677	14.752000	14.706395
1.000000	1 500000	15.520833	15.572800	15.547921
1.000000	1.600000	16.360000	16.390100	16.391150
1 000000	1.700000	17 199167	17.203300	17.232766
1 000000	1.800000	18.033333	18.011800	18.048636
1 000000	1 900000	18.857500	18 815000	18.875552
1.000000	2 000000	19 666667	19.612400	19.717749
1.000000	2 100000	20 457167	20.403300	20.427167
1.000000	2.200000	21.230667	21.187200	21.215274
1.000000	2.300000	21.990167	21.963600	22.989935
1.000000	2 400000	22.738677	22.732100	22.736344
1.000000	2 500000	23 479167	23.492200	23.483185
1 000000	2 600000	24 214667	24.243400	24.299485
1.000000	2 700000	24.948167	24.985300	24.964167
1.000000	2.800000	25.682667	25.717500	25.667085
1.000000	2.900000	26.421167	26.439700	26.417486
1.000000	3 000000	27.166667	27.151500	26.945260
1.100000	1.000000	11.500000	11.446900	11.430027
1.100000	1.100000	12.264167	12.275200	12.269364
1.100000	1.200000	13.053333	13.102600	13.044518
1.100000	1.300000	13.862500	13.928400	13.871958
1.100000	1.400000	14.686677	14.752000	14.708116
1.100000	1.500000	15.520833	15.572800	15.549450
1.100000	1.600000	16.360000	16.390200	16.392507
1.100000	1.700000	17.199167	17.203400	17.233973
1.100000	1.800000	18.033333	18.011900	18.030717
1.100000	1.900000	18.857500	18.815200	18.899829
1.100000	2.000000	19.666667	19.612500	19.718640
1.100000	2.100000	20.457167	20.403500	20.424773
1.100000	2.200000	21.230667	21.187400	21.216083
1.100000	2.300000	21.990167	21.963900	21.990731
1.100000	2.400000	22.738677	22.732400	22.747147
1.100000	2.500000	23.479167	23.492500	23.514025
1.100000	2.600000	24.214667	24.243700	24.300312

1 100000	2 700000	24 948167	24 985600	24 995192
1 100000	2 800000	25 682667	25 717900	25 668068
1 100000	2 900000	26 421167	26 440100	26 418542
1 100000	3 000000	27 166667	27 152000	26 946396

Table 1 the result comparison

References

1 Faux, I D and Pratt, M J. Computational Geometry for Computer Aided Design and Manufacturing. Ellis Horwood, UK (1978).

2 Mortenson,M E. Geometric Modeling. John Wiley (1985).

3 Lee, Y S and Chang, T C. CASCAM -an automated system for sculptured surface cavity machining. Compute. Indust. VOL. 16 (1991) PP 321-342.

4 Wickers, G W and Bradley, C. Curve and surface machining through circular are interpolation. Compute. Indust. VOL. 19 (1992) PP 329-337.

5 Piegl, L and TIller, W. Curve and surface constructions using rational B-splines. Compute. Aided Des. VOL. 19 (1987) PP 485-498.

6 Boehm, W. A survey of curve and surface methods in CAGD. Compute. -Aided Geom. Design VOL. 1 (1984) PP 1-60.

7 Hermann, G. Free-form shapes:an integrated CAD/CAM system. Compute. Indust. VOL. 5 (1984) PP 205-210.

8 Piegl, L. Key development in Computer-aided geometric design. Compute-Aided Design VOL. 1 (1984) PP 262-274.

9 P, Gu and X, Yan. Neural network approach to the reconstruction of freeform surfaces for reverse engineering. Compute. -Aided Design VOL. 27 (1995) PP 59-64.

10 Jiao Licheng , The Neural Network System Theories ,The Xian Electronic Science And Technology University , PP 101-106.

11 Chen, Y D, Tang, X J, Ni, J and Wu, S M. Automatic digitization of freeform curves by coordinate measuring machines. ASME PED-VOL 62 (1992) PP 113-125.

12 Saito, K and Miyoshi, T. Non-contact 3-D digitizing and machining system for freeform surfaces. Ann. CIRP VOL. 40 NO 1 (1991) PP 483-486.

Surface Patches for Filling Arbitrary Topological Networks

Zhang Wenzu*, Wang Fengyin**, Wu Bo*, Yang Shuzi*
*School of Mechanical Engineering, Huazhong University of Science and Technology
Wuhan 430074, P. R. China
**Gintic Institute of Manufacturing Technology, Nanyang Technological University
71 Nanyang Drive, Singapore 638075

ABSTRACT

One method for constructing surface patches, which are used to fill arbitrary rectangular networks smoothly, is proposed in the paper. Mathematical models of two types of such surface patches are presented, one with boundaries being rational quadric and the other with boundaries being cubic Bezier curves.

KEYWORDS

Geometric Modeling, Rational Bezier Patch, Gregory Patch, G^1 Continuity

1. INTRODUCTION

It is often needed to construct n-side surface patches during a shape design. A n-side surface patch can be divided into n rectangular surface patches. Through such dividing, filling arbitrary topological networks with n-side patches can be transformed into filling arbitrary rectangular networks with rectangular surface patches. The G^1 continuity between adjacent rectangular patches is the key of this problem. When Bezier patches are used to fill arbitrary rectangular networks, determining its interior control points is difficult because of the intertwinement among these points, the intertwinement will occur to guarantee the G^1 continuity conditions. The Gregory patch, introduced by Chiyokura & Kimura [1], can be used to fill arbitrary rectangular networks because no intertwinement will occur among the interior control points when G^1 continuity between any two adjacent patches is guaranteed.

For the nominal shapes of mechanical parts, there are many surface patches whose boundaries are rational Bezier curves, such as arcs and other quadric curves. In designing these shapes, such rectangular patches must be constructed to fill arbitrary rectangular networks. In this project, two types of surface patches analogous to Gregory patch were considered. In this paper, the mathematical models are given in terms of the explicit G^1 continuity conditions of adjacent rational Bezier patches. The boundaries of the two types of patches are rational quadric and cubic Bezier curves, respectively. In section 2, explicit G^1 continuity for adjacent rational Bezier patches will be introduced using the transitivity of geometric continuity. In section 3 and 4, mathematical models of the two types of surface patches will be developed using explicit continuity conditions.

2. EXPLICIT G^1 CONTINUITY CONDITIONS FOR ADJACENT RATIONAL BEZIER PATCHES

Let $P(u,v)$ and $P^*(u,v)$ be two rational Bezier patches: $P(u,v)= T(u,v)/\Omega(u,v)$, $P^*(u,v)=T^*(u,v)/\Omega^*(u,v)$. $T(u,v)$, $\Omega(u,v)$, $T^*(u,v)$ and $\Omega^*(u,v)$ can be expressed to be:

$$T(u,v) =\sum_{i=0}^{m}\sum_{j=0}^{n} Q_{ij}\,\omega_{ij}\,B_{i,m}(u)\,B_{j,n}(v), \qquad \Omega(u,v) =\sum_{i=0}^{m}\sum_{j=0}^{n} \omega_{ij}\,B_{i,m}(u)\,B_{j,n}(v)$$

$$T^*(u,v) = \sum_{i=0}^{m'}\sum_{j=0}^{n} Q_{ij}^*\,\omega_{ij}^*\,B_{i,m'}(u)\,B_{j,n}(v), \qquad \Omega^*(u,v) =\sum_{i=0}^{m'}\sum_{j=0}^{n} \omega_{ij}^*\,B_{i,m'}(u)\,B_{j,n}(v)$$

where $\{Q_{ij}\}$, $\{Q_{ij}^*\}$ and $\{\omega_{ij}\}$, $\{\omega_{ij}^*\}$ are control points and the corresponding weights, and $B_{i,m}(u)$, $B_{j,n}(v)$, $B_{i,m'}(u)$ are all Bernstein polynomials.

Assume that $P(u,v)$ and $P^*(u,v)$ have the same boundary $P(1,v) = P^*(0,v)$, where $Q_{mj} = Q_{0j}^*$ ($j=0, ..., n$), and the G^1 continuity along the common boundary is required. The necessary and sufficient condition for the G^1 continuity can be obtained as [2,3]

$$|X_0 \ X_1 \ D_1X \ D_1X^*|=0 \qquad (2.1)$$

where

$$X_0 = (C_0 \ \Omega_0)^T, \qquad X_1 = (C_1 \ \Omega_1)^T,$$

$$D_1X = (D_1C \ D_1\Omega)^T, \qquad D_1X^* = (D_1C^* \ D_1\Omega^*)^T,$$

$$D_1C = \partial T(u,v)/\partial u \,|_{u=1}, \qquad D_1\Omega = \partial\Omega(u,v)/\partial u \,|_{u=1},$$

$$D_1C^* = \partial T^*(u,v)/\partial u \,|_{u=0}, \qquad D_1\Omega^* = \partial\Omega^*(u,v)/\partial u \,|_{u=0},$$

$$C_0=\sum_{j=0}^{n-1} Q_{mj}\,\omega_{mj}\,B_{j,n-1}(v), \qquad \Omega_0=\sum_{j=0}^{n-1} \omega_{mj}\,B_{j,n-1}(v),$$

$$C_1=\sum_{j=1}^{n} Q_{mj}\omega_{mj}B_{j-1,n-1}(v), \qquad \Omega_1=\sum_{j=1}^{n} \omega_{mj}B_{j-1,n-1}(v).$$

Assume that there are two rational Bezier surface patches, $S(u,v)$ and $S^*(u,v)$. Each has one boundary which coincides with the common boundary of $P(u,v)$ and $P^*(u,v)$, i.e., $P(1,v) = P^*(0,v) = S(u,v) = S^*(0,v)$. According to the transitivity of geometrical continuity[4], if the G^1 continuity between two surface patches $P(u,v)$ and $S^*(u,v)$, $S(u,v)$ and $S^*(u,v)$, $S(u,v)$ and $P^*(u,v)$, is guaranteed, the G^1 continuity between $P(u,v)$ and $P^*(u,v)$ can be guaranteed. The explicit continuity conditions[5] are deduced below.

2.1 G^1 Continuity Condition Between P(u,v) and S*(u,v)

Let D_1Y^* be the counterpart of $S^*(u,v)$, like D_1X^* of $P^*(u,v)$. From (2.1) we can obtain

$$D_1X = p_0 X_0 + p_1 X_1 + p_2 D_1Y^* \qquad (2.2)$$

where p_i(i=0, 1, 2) are polynomials with real coefficients.

2.2 G^1 Continuity Condition Between S(u,v) and S*(u,v)

Similar to Eq. (2.2), we have $D_1Y^* = r_0 X_0 + r_1 X_1 + r_2 D_1Y$, where D_1Y is the counterpart of $S(u,v)$, like D_1X of $P(u,v)$. Let $r_0 = r_1 = 0$ and $r_2 = 1$, we can obtain

$$D_1Y^* = D_1Y \qquad (2.3)$$

2.3 G^1 Continuity Condition Between S(u,v) and P*(u,v)

Similar to Eq. (2.2), we have

$$D_1X^* = q_0 X_0 + q_1 X_1 + q_2 D_1Y \qquad (2.4)$$

where q_i(i = 0, 1, 2) are polynomials with real coefficients. Let $D_1Y=d$, a polynomial vector with real coefficients. From (2.2), (2.3) and (2.4), the explicit continuity conditions are obtained as

$$D_1X = p_0 X_0 + p_1 X_1 + p_2 d \qquad (2.5)$$

$$D_1X^* = q_0 X_0 + q_1 X_1 + q_2 d \qquad (2.6)$$

3. SURFACE PATCHES WITH BOUNDARIES OF RATIONAL QUADRIC BEZIER CURVES

3.1 G^1 Continuity Conditions Between Two Adjacent Rational Quadric Bezier Patches

Let $P(u,v)$ and $P^*(u,v)$ be two 2×2 rational Bezier patches, and

$$D_1X = 2\sum_{i=0}^{2} a_i B_{i,2}(v), \qquad D_1X^* = 2\sum_{i=0}^{2} b_i B_{i,2}(v)$$

$$X_0=\sum_{i=0}^{1} Z_i B_{i,1}(v), \qquad X_1=\sum_{i=1}^{2} Z_i B_{i-1,1}(v)$$

where $a_i = (Q_{2i}\omega_{2i}-Q_{1i}\omega_{1i} \ \ \omega_{2i}-\omega_{1i})^T$, $b_i = (Q_{1i}^*\omega_{1i}^*-Q_{2i}\omega_{2i} \ \ \omega_{1i}^*-\omega_{2i})^T$, $Z_i = (Q_{2i}\omega_{2i} \ \ \omega_{2i})^T$.

Let $d = \sum_{i=0}^{1} d_i B_{i,1}(v)$, $p_k = 2\sum_{i=0}^{1} p_{ki} B_{i,1}(v)$, $q_k = 2\sum_{i=0}^{1} q_{ki} B_{i,1}(v)$, (k=0, 1, 2), in which p_{ki} and q_{ki} are real numbers, d_i are real vectors. Substituting d and p_k into (2.5), we obtain

$$a_0 = p_{00} Z_0 + p_{10} Z_1 + p_{20} d_0 \quad (3.1)$$

$$a_1 = (p_{01} Z_0 + p_{00} Z_1 + p_{10} Z_2 + p_{11} Z_1 + p_{20} d_1 + p_{21} d_0)/2 \quad (3.2)$$

$$a_2 = p_{01} Z_1 + p_{11} Z_2 + p_{21} d_1 \quad (3.3)$$

Substituting d and q_k into (2.6), we obtain

$$b_0 = q_{00} Z_0 + q_{10} Z_1 + q_{20} d_0 \quad (3.4)$$

$$b_1 = (q_{01} Z_0 + q_{00} Z_1 + q_{10} Z_2 + q_{11} Z_1 + q_{20} d_1 + q_{21} d_0) / 2 \quad (3.5)$$

$$b_2 = q_{01} Z_1 + q_{11} Z_2 + q_{21} d_1 \quad (3.6)$$

Assume that the boundaries of P(u,v) and $P^*(u,v)$ are given and d_0 and d_1 are set as such that $|a_0\ Z_0\ Z_1\ d_0| = 0$, $|b_0\ Z_0\ Z_1\ d_0| = 0$, $|a_2\ Z_1\ Z_2\ d_1| = 0$, $|b_2\ Z_1\ Z_2\ d_1| = 0$. The values of p_{ki} and q_{ki} (k=0, 1, 2, i=0,1) can be determined from (3.1), (3.3), (3.4) and (3.6) when d_0 and d_1 are determined. The vectors a_1 and b_1, which reflect the positions of interior control points, can be determined from (3 2) and (3.5).

Suppose that $d_0 = (Q_d\omega_d\ \omega_d)^T$. Substituting d_0 into $|a_0\ Z_0\ Z_1\ d_0| = 0$, we obtain $|Q_{20}\text{-}Q_{10}\ Q_{21}\text{-}Q_{20}\ Q_d\text{-}Q_{20}| = 0$. Substituting d_0 into $|b_0\ Z_0\ Z_1\ d_0| = 0$, we obtain $|Q_{10}^*\text{-}Q_{20}\ Q_{21}\text{-}Q_{20}\ Q_d\text{-}Q_{20}| = 0$. If the three vectors, $(Q_{20}\text{-}Q_{10})$, $(Q_{10}^*\text{-}Q_{20})$ and $(Q_{21}\text{-}Q_{20})$, are on the same plane and Q_d is on the plane, too, conditions $|a_0\ Z_0\ Z_1\ d_0| = 0$ and $|b_0\ Z_0\ Z_1\ d_0| = 0$ will be guaranteed. In general, we take $Q_d = (Q_{10}+Q_{10}^*)/2$ and $\omega_d = 1$.

3.2 Surface Patches With Boundaries of Rational Quadric Bezier Curves

Let us consider a surface patch with boundaries of rational quadric Bezier curves now. In order to guarantee G^1 continuity between one surface patch and every other surface patch around it, four interior control points are needed to control, respectively, the cross-boundary derivatives, in a homogeneous coordinate system, as presented in section 3.1. A new surface patch can then be represented as:

$$S(u,v) = [\sum_{i=0}^{2}\sum_{j=0}^{2} Q_{ij}(u,v)\, \omega_{ij}(u,v)\, B_{i,2}(u)\, B_{j,2}(v)] / [\sum_{i=0}^{2}\sum_{j=0}^{2} \omega_{ij}(u,v)\, B_{i,2}(u)\, B_{j,2}(v)]$$

where

$Q_{11}(u,v) = [v(1\text{-}v)(1\text{-}u)\, Q_{11}^{u0} + v(1\text{-}v)u\, Q_{11}^{u1} + u(1\text{-}u)(1\text{-}v)\, Q_{11}^{v0} + u(1\text{-}u)v\, Q_{11}^{v1}] / [u(1\text{-}u)+v(1\text{-}v)]$,

$Q_{ij}(u,v) = Q_{ij}$, $(i\neq1, j\neq1)$,

$\omega_{11}(u,v) = [v(1\text{-}v)(1\text{-}u)\, \omega_{11}^{u0} + v(1\text{-}v)u\, \omega_{11}^{u1} + u(1\text{-}u)(1\text{-}v)\, \omega_{11}^{v0} + u(1\text{-}u)v\, \omega_{11}^{v1}] / [u(1\text{-}u)+v(1\text{-}v)]$,

$\omega_{ij}(u,v) = \omega_{ij}$, $(i\neq1, j\neq1)$,

where $\{Q_{ij}\}$ and $\{\omega_{ij}\}$ are control points and the corresponding weights on boundaries, Q_{11}^{u0}, Q_{11}^{u1}, Q_{11}^{v0}, Q_{11}^{v1} are the interior control points, and ω_{11}^{u0}, ω_{11}^{u1}, ω_{11}^{v0}, ω_{11}^{v1} are the corresponding weights.

This surface patch's four boundaries are rational quadric Bezier curves. The cross-boundary derivatives of the four boundaries are influenced by four interior control points and their corresponding weights, respectively.

4. SURFACE PATCHES WITH BOUNDARIES OF RATIONAL CUBIC BEZIER CURVES

4.1 G^1 Continuity Conditions Between Two Adjacent Rational Cubic Bezier Patches

Let P(u,v) and $P^*(u,v)$ be two rational 3×3 Bezier patches, and

$$D_1X = 3\sum_{i=0}^{3} a_i B_{i,3}(v), \qquad D_1X^* = 3\sum_{i=0}^{3} b_i B_{i,3}(v)$$

$$X_0=\sum_{i=0}^{2} Z_i B_{i,2}(v), \qquad X_1=\sum_{i=1}^{3} Z_i B_{i-1,2}(v)$$

where $a_i=(Q_{3i}\omega_{3i}-Q_{2i}\omega_{2i}\ \ \omega_{3i}-\omega_{2i})^T$, $b_i=(Q_{1i}^{*}\omega_{1i}^{*}-Q_{3i}\omega_{3i}\ \ \omega_{1i}^{*}-\omega_{3i})^T$, $Z_i=(Q_{3i}\omega_{3i}\ \ \omega_{3i})^T$

Let $d=\sum_{i=0}^{2} d_i B_{i,2}(v)$, $p_k=3\sum_{i=0}^{1} p_{ki} B_{i,1}(v)$, $q_k=3\sum_{i=0}^{1} q_{ki} B_{i,1}(v)$, $(k=0, 1, 2)$, where p_{ki} and q_{ki} are real numbers and d_i are real vectors. Substituting d and p_k into (2.5), we obtain

$$a_0=p_{00}Z_0+p_{10}Z_1+p_{20}d_0 \qquad (4.1)$$
$$a_1=(2p_{00}Z_1+p_{01}Z_0+2p_{10}Z_2+p_{11}Z_1+2p_{20}d_1+p_{21}d_0)/3 \qquad (4.2)$$
$$a_2=(2p_{01}Z_1+p_{00}Z_2+2p_{11}Z_2+p_{10}Z_3+2p_{21}d_1+p_{20}d_2)/3 \qquad (4.3)$$
$$a_3=p_{01}Z_2+p_{11}Z_3+p_{21}d_2 \qquad (4.4)$$

By substituting d and q_k into (2.6), we obtain

$$b_0=q_{00}Z_0+q_{10}Z_1+q_{20}d_0 \qquad (4.5)$$
$$b_1=(2q_{00}Z_1+q_{01}Z_0+2q_{10}Z_2+q_{11}Z_1+2q_{20}d_1+q_{21}d_0)/3 \qquad (4.6)$$
$$b_2=(2q_{01}Z_1+q_{00}Z_2+2q_{11}Z_2+q_{10}Z_3+2q_{21}d_1+q_{20}d_2)/3 \qquad (4.7)$$
$$b_3=q_{01}Z_2+q_{11}Z_3+q_{21}d_2 \qquad (4.8)$$

As has been analyzed in section 4.1, d_0 and d_1 must be determined to satisfy the following relations: $|a_0\ Z_0\ Z_1\ d_0|=0$, $|b_0\ Z_0\ Z_1\ d_0|=0$, $|a_3\ Z_2\ Z_3\ d_2|=0$, $|b_3\ Z_2\ Z_3\ d_2|=0$. In general, d_1 can be set as $(d_0+d_2)/2$, a_1, a_2, b_1 and b_2 are determined from (4.2), (4.3), (4.6) and (4.7), respectively.

4.2 Surface Patches With Boundaries of Rational Cubic Bezier Curves

In order to guarantee G^1 continuity between the surface patch and every other surface patch around it, interior control points and the corresponding weights need to be increased so that the intertwinement among these points is removed easily according to the results of section 4.1. A new surface patch can be represented as

$$S(u,v)=\left[\sum_{i=0}^{3}\sum_{j=0}^{3} Q_{ij}(u,v)\,\omega_{ij}(u,v)\,B_{i,3}(u)\,B_{j,3}(v)\right]\Big/\left[\sum_{i=0}^{3}\sum_{j=0}^{3}\omega_{ij}(u,v)\,B_{i,3}(u)\,B_{j,3}(v)\right]$$

where

$Q_{11}(u,v)=(u\,Q_{11}^{v0}+v\,Q_{11}^{u0})/(u+v)$,

$\omega_{11}(u,v)=(u\,\omega_{11}^{v0}+v\,\omega_{11}^{u0})/(u+v)$,

$Q_{12}(u,v)=[u\,Q_{12}^{v1}+(1-v)\,Q_{12}^{u0}]/(u+1-v)$,

$\omega_{12}(u,v)=[u\,\omega_{12}^{v1}+(1-v)\,\omega_{12}^{u0}]/(u+1-v)$,

$Q_{21}(u,v)=[(1-u)\,Q_{21}^{v0}+v\,Q_{21}^{u1}]/(1-u+v)$,

$\omega_{21}(u,v)=[(1-u)\,\omega_{21}^{v0}+v\,\omega_{21}^{u1}]/(1-u+v)$,

$Q_{22}(u,v)=[(1-u)\,Q_{22}^{v1}+(1-v)\,Q_{22}^{u1}]/(2-u-v)$,

$\omega_{22}(u,v)=[(1-u)\,\omega_{22}^{v1}+(1-v)\,\omega_{22}^{u1}]/(2-u-v)$.

$Q_{ij}(u,v)=Q_{ij}$, $\omega_{ij}(u,v)=\omega_{ij}$.

For increasing the ability of modifying the interior of the surface patch, its representation can be further modified. The interior control points and the corresponding weights can be changed to be

$Q_{11}(u,v)=(u^2Q_{11}^{v0}+2uv\,Q_{11}+v^2Q_{11}^{u0})/(u+v)^2$,

$\omega_{11}(u,v)=(u^2\omega_{11}^{v0}+2uv\,\omega_{11}+v^2\omega_{11}^{u0})/(u+v)^2$,

$Q_{12}(u,v)=[u^2Q_{12}^{v1}+2u(1-v)\,Q_{12}+(1-v)^2Q_{12}^{u0}]/(u+1-v)^2$,

$\omega_{12}(u,v)=[u^2\omega_{12}^{v1}+2u(1-v)\,\omega_{12}+(1-v)^2\omega_{12}^{u0}]/(u+1-v)^2$,

$Q_{21}(u,v)=[(1-u)^2Q_{21}^{v0}+2(1-u)v\,Q_{21}+v^2Q_{21}^{u1}]/(1-u+v)^2$,

$\omega_{21}(u,v)=[(1-u)^2\omega_{21}^{v0}+2(1-u)v\,\omega_{21}+v^2\omega_{21}^{u1}]/(1-u+v)^2$,

$Q_{22}(u,v)=[(1-u)^2Q_{22}^{v1}+2(1-u)(1-v)\,Q_{22}+(1-v)^2Q_{22}^{u1}]/(2-u-v)^2$,

$\omega_{22}(u,v) = [(1-u)^2 \omega_{22}^{v1} + 2(1-u)(1-v)\,\omega_{22} + (1-v)^2 \omega_{22}^{u1}] / (2-u-v)^2$. In the above formulae, Q_{ij}, Q_{11}^{v0}, Q_{11}^{u0}, Q_{12}^{v1}, Q_{12}^{u0}, Q_{21}^{v0}, Q_{21}^{u1}, Q_{22}^{v1}, Q_{22}^{u1} are control points and ω_{ij}, ω_{11}^{v0}, ω_{11}^{u0}, ω_{12}^{v1}, ω_{12}^{u0}, ω_{21}^{v0}, ω_{21}^{u1}, ω_{22}^{v1}, ω_{22}^{u1} are their corresponding weights.

5. CONCLUSION

Explicit continuity conditions for adjacent two rational Bezier patch are deduced according to the transitivity of geometrical continuity. The mathematical models of two types of surface patches whose boundaries are rational Bezier curves are given in the paper. Arbitrary rectangular topological networks can be filled with such patches and G^1 continuity among adjacent patches can be ensured easily.

REFERENCE

[1] Chiyokura, F. Kimura, Design of Solids with Free-form Surfaces, Computer Graphics, 17(1983): 289~298.

[2] DeRose, Necessary and Sufficient Conditions for Tangent Plane Continuity of Bezier Surface, CAGD, 7(1990): 165~179.

[3] Liu, GC^1 Conditions Between Two Adjacent Rational Bezier Surface Patches, CAGD, 7(1990): 151~163.

[4] Shirman, C. Sequin, Local Surface Interpolation with Shape Parameters Between Adjoining Gregory Patches, CAGD, 7(1990):375~388.

[5] Degen, Explicit Continuity Conditions for Adjacent Bezier Surface Patches, CAGD, 7(1990): 187~190.

CONSTRAINT-BASED PLANAR FREE FORM SHAPE

T.W. Lam[a], K.M. Yu[b] and Y.C. Yip[a]
[a]Department of Mechanical Engineering, The Hong Kong Polytechnic University, Hong Kong.
[b]Department of Manufacturing Engineering, The Hong Kong Polytechnic University, Hong Kong.

ABSTRACT

In practical engineering design, shape design is iteration-based, i.e. designers start with a sketch and refine it until it meets their requirements. In order to release designers to concentrate on the creative part of design, a constraint-based variational geometry technique is used to support high-level shape relationship and to enhance model reusability. Since the nonuniform rational B-spline (NURBS) description has become the standard geometry description in the field of computer aided design (CAD), the creation and manipulation of complex geometrical models would be impossible without NURBS. The popularity of NURBS in CAD systems has led to an increasing research on shape variation of NURBS. However, the previous works were focused on the quantitative variation of shape parameters without any link to incorporate the engineering dimension constraint. In this paper, the authors' work on dimension-driven variation of NURBS based planar shape is presented.

KEYWORDS

NURBS, Variational Geometry, Constraint-based Design

1. INTRODUCTION

In practical engineering design, shape design is typically accomplished in an interactive and iterative manner such that designers start with a sketch and refine it until it meets their requirements. Shape variations have been of interest for more than two decades. Variational geometry is a key technique to synthesize shape on the basis of designers' intentions. This is achieved by specifying the geometric and engineering shape constraints which, when solved, define the geometry of the object.

The published variation techniques are depended on the underlying representation scheme. Lin, Light and Anderl [1, 2, 3] employed techniques for shapes composed of straight lines and circular arcs. Emphasis were on devising efficient constraint representation and satisfaction schemes. Beaty et al. had worked on including more complex edges comprising cubic Hermite and Bezier curves in variational geometry [4]. Extension to cover conics by using rational Bezier curves was also reported [5].

Since the NURBS description has become the standard curve and surface description in the field of CAD, the creation and manipulation of complex geometrical models would be impossible without NURBS as they can represent ordinary splines as well as all conics and composite curves. The popularity of NURBS in CAD systems has led to increasing research on NURBS applications in shape variation. For instance, Lane [6] and Nahas [7] accomplished spline modifications by refining and moving control points. Refinement can also be made by either degree elevation or knot insertion where modifications can be restricted to a certain segment of the curve. For instance, Coons used special blending functions such as tensioned splines [8] while Barsky applied the concept of geometric continuity [9]. The relaxation of parametric continuity, however, introduces free parameters, e.g. tangent magnitude or curvature controllers. Versprille [10] initiated the investigation of the effects of the weights of control points on the uniform rational B-spline surface shape. Piegl [11, 12] proposed two methods to vary the shape of NURBS curves: interactive control points location-based and weight-based modification. He also quantified the push/pull effect of weights for NURBS curves and summarized all the important characteristics, advantages and properties of NURBS to provide a comprehensive set of fundamental algorithms necessary to design and visualize the NURBS defined geometry. Bartels [13] proposed a derivative preserving method to control the position, tangency and curvature of a B-spline curve by using control points. These works, however, focused on the quantitative variation of shape parameters without any link to incorporate the engineering dimension constraints. In this paper, the authors' work on the dimension-driven variation of NURBS based planar shape is thus presented.

2. CONSTRAINT-BASED PLANAR NURBS SHAPE DESIGN

For engineering shape design, there are plenties of interacting dimensions. The constraint-based design technique is used to capture designers' intentions into a form of interacting dimensional relationships. In the proposed constraint-based planar free form shape modeling system, the dimension constraints are applied on the control polygons rather than the NURBS curve. This then provides an intuitive design means. It is because one characteristic of the NURBS curve is to follow the shape of control polygon. As the control polygon is composed of a series of line segments, the descriptive elements of a control polygon are thus:

1. position of control vertex
2. length of segment of control polygon
3. orientation of segment of control polygon
4. angle between segments of control polygon
5. ratio of chord length and amplitude of control polygon

The first three constraints are the natural means to control the position and orientation of a line segment. The fourth constraint is used to control the turning of the NURBS curve. The fifth constraint is used to control the bulge factor of the NURBS curve. The chord length is defined as the distance between two specified control points of the control polygon. The amplitude is defined as the maximum perpendicular distance from a control point to the chord as shown in Figure 1. With the captured dimension constraints on the control polygon, a constraint network is formed to control the propagation of design modification on the NURBS curve.

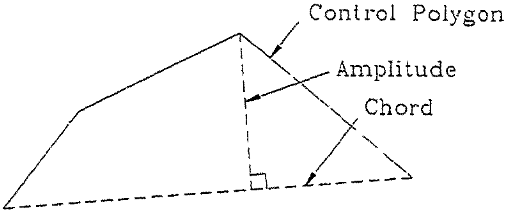


Figure 1 Definitions of chord and amplitude

3. CONSTRAINTS

The constraints are grouped into two categories: constraint_single and constraint_multiple. The constraint_single constraints are used to represent the dimensional constraints on a single line segment. The constraint_multiple constraints are used to capture the dimensional relationships between multiple line segments. Since the position, i.e. X- and Y-coordinates, of control vertex, length and orientation of line segment are descriptive elements for controlling a single line segment, the constraint_single constraints as listed in Table 1 are thus established.

Name of constraint	Description of constraint
ANGLE_FIX	Orientation of line segment is fixed.
ANGLE_RANGE	Orientation of line segment is within specified range.
LENGTH_FIX	Length of line segment is fixed.
LENGTH_RANGE	Length of line segment is within specified range.
FIX_X	X-coordinate of an end point of line segment is fixed.
FIX_Y	Y-coordinate of an end point of line segment is fixed.
FIX_XY	Both X- and Y-coordinates of an end point of line segment are fixed.

Table 1 Constraint_single constraints

In order to provide intuitive means for controlling the turning and bulge factor of the NURBS curve, the constraint_multiple constraints as listed in Table 2 are used.

Name of constraint	Description of constraint
CHORD_LENGTH_FIX	Ratio of chord length and amplitude is fixed.
CHORD_LENGTH_RANGE	Ratio of chord length and amplitude is within specified range.
AMP_RATIO_FIX	Ratio of two amplitudes for two selected chords is fixed.
AMP_RATIO_RANGE	Ratio of two amplitudes for two selected chords is within specified range.
ANGLE_TWO_LINE	Angle between two consecutive line segments is fixed.
ANGLE_CONVEX	Angle between two consecutive line segments is acute.
ANGLE_CONCAVE	Angle between two consecutive line segments is obtuse.

Table 2 Constraint_multiple constraints

A sample application of the developed constraints for the shape design of a facial profile and its accompanying constraint network are shown respectively in Figure 2 and Figure 3.

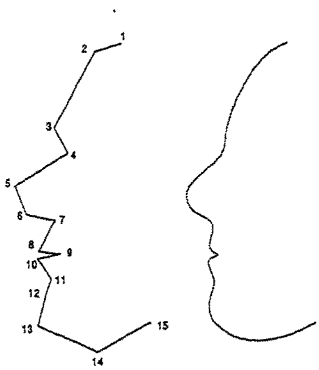


Figure 2 Facial profile

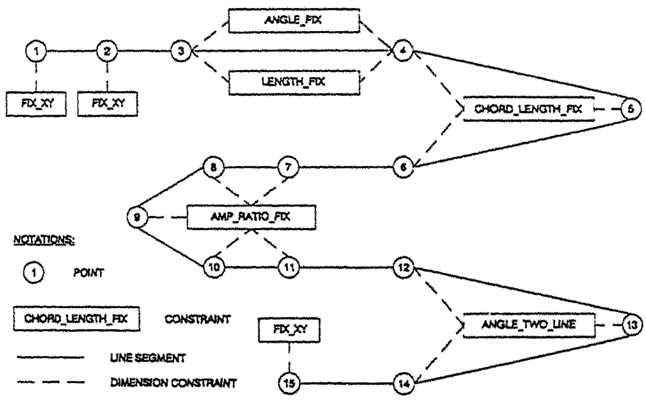


Figure 3 Constraint network of facial profile

4. CONSTRAINT MANAGEMENT SYSTEM

A constraint management system is developed for manipulating design modification on the constrained NURBS curve. The modification of the NURBS curve is activated by changing a dimension constraint, such as the position of a control vertex, the length of a line segment, the orientation of a line segment, the ratio of the chord length and amplitude, etc. The control polygon is then modified according to its captured constraints. As the control polygon is composed of a series of line segments, the basic problem in modifying the control polygon is to determine of the geometric parameters of the line segment, i.e. X- and Y-coordinates of starting point, X- and Y-coordinates of the end point, length and orientation of the line segment.

The objective of the constraint management system is to solve all the geometric parameters of all line segments that satisfies the modified dimension constraint and the captured constraints. The effect of changing the dimension constraint is propagated to other control points by traversing through the constraint network. A case matching and solving mechanism is used to implement the constraint management system.

By listing all possible combinations of constraining the geometric parameters of a line segment, as shown in Table 3, the method for evaluating the unknown geometric parameters of the line segment can be identified by a simple table look up. Before looking up the table, the coordinates of either the starting point or the end point of the line segment should be known. Otherwise, the geometric parameters of the line segment cannot be solved. In Table 3, it is assumed that the coordinates of the starting point of the line segment are known.

Case	P_1	P_2	P_3	P_4	P_5	P_6	Action to be taken
1	✔	✔	✔	✔	✔	✔	Check: Orientation and length are conformed with the starting point and the end point.
2	✔	✔	✔	✔	✔		Check: Orientation is conformed with the starting point and the end point.
3	✔	✔	✔	✔		✔	Check: Length is conformed with the starting point and the end point.
4	✔	✔	✔	✔			Already solved
5	✔	✔	✔		✔	✔	Find: X- and Y-coordinates of the end point Check: Above determined X-coordinate is conformed with the constrained X-coordinate.
6	✔	✔	✔		✔		Find: Y-coordinate of the end point and length
7	✔	✔	✔			✔	Find: Y-coordinate of the end point and orientation
8	✔	✔	✔				Cannot be solved: Under-constraint
9	✔	✔		✔	✔	✔	Find: X- and Y-coordinates of the end point Check: Above determined Y-coordinate is conformed with the constrained Y-coordinate.
10	✔	✔		✔	✔		Find: X-coordinate of the end point and length
11	✔	✔		✔		✔	Find: X-coordinate of the end point and orientation
12	✔	✔		✔			Cannot be solved: Under-constraint
13	✔	✔			✔	✔	Find: X- and Y-coordinates of the end point
14	✔	✔			✔		Cannot be solved: Under-constraint
15	✔	✔				✔	Cannot be solved: Under-constraint
16	✔	✔					Cannot be solved: Under-constraint

Table 3 All combinations of the constraining geometric parameters of a line segment

In Table 3, The following notations are used:

P_1 and P_2: X- and Y-coordinates of the starting point of the line segment
P_3 and P_4: X- and Y-coordinates of the end point of the line segment
P_5: Orientation of the line segment
P_6: Length of the line segment
✔: The geometric parameter is constrained.

In traversing the constraint network of the modified control polygon, the new position of the control vertices can be determined by identifying the matching case in Table 3. For instance, in order to create a bigger nose for the facial profile in Figure 2, the dimension constraint CHORD_LENGTH_FIX, being applied to control points 4, 5 and 6, is increased from 1.8 to 2.5. A new facial profile with a bigger nose is shown in Figure 4. In order to obtain a more distinct mouth, the dimension constraint AMP_RATIO_FIX, being applied to control points 7, 8, 9, 10 and 11, is increased from 0.5 to 0.8, as shown in Figure 5. In order to obtain a more angular lower jaw, the dimension constraint ANGLE_TWO_LINE, being applied to control points 12, 13 and 14, is decreased from 100 to 80, as shown in Figure 6.

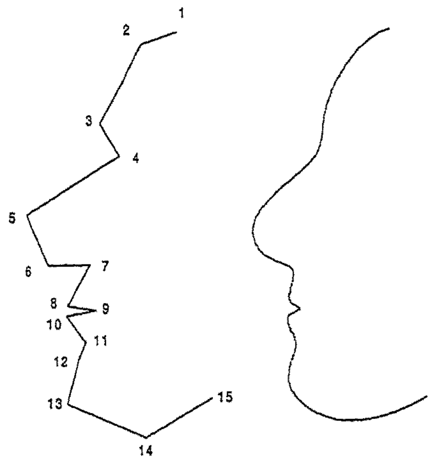


Figure 4 CHORD_LENGTH_FIX amended

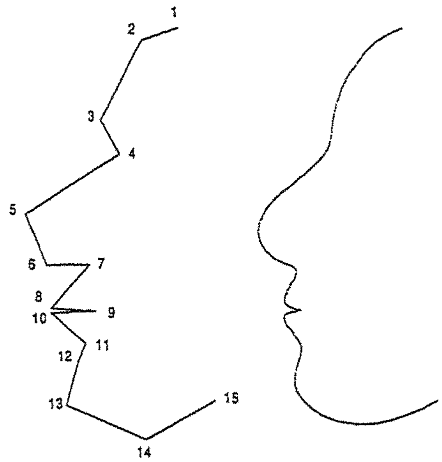


Figure 5 AMP_RATIO_FIX amended

5. CONCLUSIONS

A constraint-based planar NURBS shape design technique is presented. The essential concepts of the technique is to modify the shape of NURBS based planar curve by incorporating the dimension constraints. The constraint-based planar free form shape design system is implemented with the a set of constraint_single and constraint_multiple constraints. Finally, the functioning of the constraint management system is illustrated with the facial profile example.

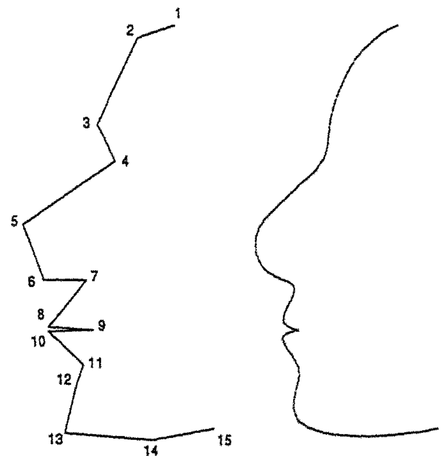


Figure 6 ANGLE_TWO_LINE amended

REFERENCES

1. Lin, V. C., Gossard, D. C. and Light, R. A., "Variational geometry in computer-aided design", Computer Graphics, v15, n3, pp. 171-179, 1981.

2. Light, R. and Gossard, D., "Modifications of geometric models through variational geometry", Computer-aided Design, v4, n4, pp. 209-214, 1982.

3. Anderl, R. and Mendgen, R., "Modelling with constraints: theoretical foundation and application", Computer-aided Design, v28, n3, pp. 155-168, 1996.

4. Beaty, P. L., Generalized Three-dimensional Variational Geometry, Master's Thesis, Colorado State University, USA, 1988.

5. Beaty, P. L., Fitzhorn, P. A. and Herron, G. J., "Extensions in variational geometry that generate and modify object edges composed of rational Bezier curves", Computer-aided Design, v26, n2, pp. 98-108, Feb 1994.

6. Lane, J. F., Shape Operators for Computer-aided Geometric Design, PhD thesis, University of Utah, USA, 1977.

7. Nahas, M., Huitric, H. and Saintourens, M., "Animation of a B-spline figure", The Visual Computer, v3, pp. 272-276, 1988.

8. Coons, S. A., "Modification of the shape of piecewise curves", Computer-aided Design, v9, n3, pp. 178-180, 1977.

9. Barsky, B. A., The Beta-spline: a Local Representation Based On Shape Parameters and Fundamental Geometric Measures, PhD thesis, University of Utah, USA, 1981.

10. Versprille, K., Computer-aided Design Applications of the Rational B-spline Approximation Form, PhD thesis, Syracuse University, USA, 1975.

11. Piegl, L., "Modifying the shape of rational B-spline. Part 1: Curves", Computer-aided Design, v21, n8, pp. 509-518, 1989.

12. Piegl, L., "Rational B-spline curves and surfaces for CAD and graphics" in State of the Art in Computer Graphics - Visualization and Modelling, Rogers, F. and Earnshaw, A. (eds.), Springer-Verlag, 1991.

13. Fowler, B. and Bartels, R., "Constraint based curve manipulation", IEEE Computer Graphics & Applications, pp. 43-49, Sep 1993.

A DESIGN SUPPORT SYSTEM FOR BASIC DESIGN OF POWER PLANT HEAT-EXCHANGER

Zhihua Zhang† **Hideyuki Yamaguchi††** **Isao Nagasawa†††**
Masanobu Umeda†††

†Graduate School of Faculty of Computer Sci. and System Eng., Kyushu Institute of Technology, Japan, email: zhang@cad.mse.kyutech.ac.jp
††Department of System Technology Developing, Choryo Design Ltd. Co., Japan
†††Faculty of Computer Sci. and System Eng., Kyushu Institute of Technology, Japan

ABSTRACT

Traditional CAD systems are good to support drafting but do not support design process itself because the systems are based on shape-oriented modeling approaches but pay no or few attention on design activities. This paper presents a design support system for the basic design of power plant heat-exchanger based on the features of design activities in the design phase. Some basic concepts are given firstly for modeling and representing design objects and design process as well as system construction, then the system constructing method is discussed. An integrated support environment are provided finally in terms of the strategy that designers can use the system actively and pay less attentions to data management.

KEYWORDS

Design Support System, Design Task, Design Calculation, Generate-and-Test

1. INTRODUCTION

The design business of shipbuilding, power plant machine, and other civil used products are mainly concerned about getting high reliability and low cost for high competition. Not only it is necessary to improve the intellectual productivity of design work, but also to give a consideration for new products development and technique inheritance problem for raw recruit designers in organization, especially in recently, in order to cope with the problems like variety of customers' needs and internationalization of design business, there has been a considerable growth of interest in intelligent CAD systems or knowledge based design support systems. Some purposebuilt design tools have been inducted to support design analysis in power plant machine design field, however, there exist some defects which make the systems be short of flexibility in dealing with the situations and are not efficient to support the design activities. Because of the complexity and the ill-structured characteristics of design knowledge, it is generally difficult to be handled in systems. Although some systems deal with the design knowledge, many of them process it in a conventional programming language, design knowledge were shut in the control description so-called variables declaration, procedure call, conditional statement and recursion, etc. even if they adopt some traditional AI approaches or ES technologies. This makes it difficult to use and maintain design knowledge in systems, especially for designers themselves. Designers are always using the systems in passive way.

On the other hand, many systems pay less attention to collaboration relationship problem between computer systems and designers. A lot of systems are developed good for problem analyzing but short of providing the interactive abilities to support design activities for designers.

To develop a knowledge based design support system, it is necessary to analyze, describe and manage the knowledge of design process and design objects about a specific design domain firstly[4]. Our research is concerned with the formal representing method about design activities and design objects; description language for design task and design process; and the collaboration relation about designers and computer support systems. This paper focus discussion on the method of developing a design support system for a specific design phase of shell-and-tube type heat-exchanger design field by considering the features of the design phase and the problems mentioned above.

2. OVERVIEW AND FEATURES OF THE BASIC DESIGN OF POWER PLANT HEAT-EXCHANGER

2.1 The Design Flow Of Heat-exchanger

Heat-exchanger is widely used in power plant machine, for example, stream generator, heating apparatus, condenser, etc. A shell-and-tube type heat-exchanger is also called multi-pipe type heat-exchanger which contains many pipes in pressure-resisting vessel and do heat transfer by convention through the temperature difference of inner fluid and outer fluid of pipes.

The main design work of a heat-exchanger can roughly divided into three phases: the conceptual design phase, the basic design phase, and the detailed design phase. During the conceptual design phase, it is to covert the given design requirements into some skeletons of physical components or functional components of a machine by type decision such that the requirements are satisfied. The work in basic design phase is to perform the performance design about inner element components, decide major components structure and evaluate the machine performance according to the specifications got from conceptual design phase. The work in detailed design phase is to determine the detail dimensions, assembly structure and assembling sequences about components from view of manufacturing aspect based on the results of basic design. Backtracking design is needed if the results are not satisfied.

2.2 An Overview of The Basic Design And Its Features

Design activities in basic design phase are mainly concerned with design space search problems. It is aim to get a design solution with high performance and low cost such that the design requirements are satisfied. Generally, design specifications got in conceptual design phase are just given in qualitative or semi-quantitative description. Designers have to determine design problems from the qualitative specification space to functional space of the design objects, then do design calculations on the settled problems by quantitative approaches, for instance, the constraints satisfaction method. The process can be mainly outlined as two aspects: (a) to concretize the vague design specifications into more detailed design tasks and settle the specific design problems by quantitative model or well-known history design case. (b) to model the concrete design problems then to perform design calculations and evaluations to get design solutions.

At present, because computer-aided tools are spare and few, in order to get design solutions as soon as possible, designers usually take a rough process to reduce design search steps and make design space much more smaller, then to perform the local detail calculations. The features of basic design can be considered as that of design activities taken by designers in problem solving. The behavior of designers in basic design is usually to take a assumption, calculation and evaluation loop on design items with trial and error feature, and these activities are performed iteratively until the design constraints are satisfied.

3. THE OVERALL FRAMEWORK OF SYSTEM

The complexity of design problems let designers take problem reduction and trial and error behavior in design process. This let us make a decision on what type of system it should be when we want to construct an environment to support design work. In order to provide optimal capabilities in system and avoid causing problem like emptying design know-how, a careful consideration must be given on the features of designers' behaviors in design and that of computer systems. This means we should cope with the problems like, what are the design activities and design objects concerned about? what jobs in design process are fit for computer systems? what jobs should be done by designers?

To concretize the design specification to detailed problems, qualitative analysis and the comprehension of design knowledge and experience of designers is employed, while the data process or iterative calculation just fit to be processed by computer-aided tools. Figure 1 shows our main ideas of a framework arranged to support the design activities in basic design phase in terms of the considerations. It consists of a model base, a knowledge base, three support tools, a process control module and the user interface.

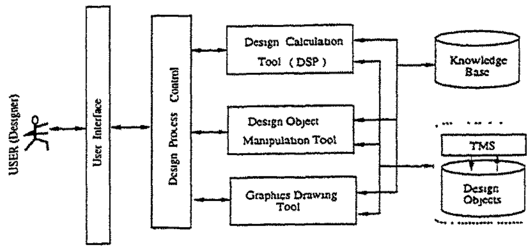


Fig. 1 The overall framework

Where, the model base store design object models generated and operated in all design steps, the knowledge base contains design knowledge which is necessary for object operation, design calculation in using the support tools, the knowledge can be decsribed by designers themselves using a specific description language. The object manipulation tool is for the edition and control of design object model. The design calculation tool is used to support design calculation activities. The graphics drawing tool is for the generation of graphics, tables, etc. as to be design documentation. The design process control module provide the abilities of overall control to support design behavior in problem reduction and design calculation.

There need two type of languages in description of design process: one is for the description of design calculation sequences and the other is for the description of design tasks. The details will be described in next section.

4. DESCRIPTION OF DESIGN PROCESS

4.1 Description of Design Calculation

The fundamental activities in a design calculation is to determine the values of required engineering variables(denoted as upper design attributes) from a set of design variables on which the required variables depend(denoted as lower design attributes), and verify the given constraints for the variables are satisfied or not. To do a local design calculation for the settled primitive calculating problems, there are several approaches for searching design solutions such as generate-and-test, inverse operation, etc. To support this by computer-aided tools in system, the assumption-calculation-evaluation activities performed by designers in small scope design calculations can be described by generate-and-test model which generate all of the solutions for certain design conditions and provide optimum selection for designers.

A specific calculating description language DSP[3] has been proposed to support the model as well as the abilities of design knowledge description. Data flow diagram is used to represent the dependency between the design variables. The detailed discussion about some basic concepts on design space, solution searching method for design calculation are presented in [2].

4.2 Description of Design Operation

Generally, it is difficult to give consideration on lots of evaluation items or design parameters at one time if the design problem is complicated. To concretize the design specifications to detailed design tasks, a design problem is often reduced into some smaller design problems which are adapt for evaluation using quantitative models or well-known history design case to satisfy the functional, structural or other design requirements, then to perform the design calculation. The decision process of problem reduction is considered as the design process performed by designers, the results can be described by a predicate-like form as following based on the rules of (a) reduction (b) selection (c) sequences.

$$\begin{array}{ll} DesignTask & \Leftarrow Guard \parallel Task_1, ..., Task_i, ..., Task_n \\ \forall\ Task_i & ::= \langle PrimitiveTask \rangle \mid \langle DesignTask \rangle \end{array}$$

where, the *Guard* is for the representation of the activities such as method selection or type decision, etc., it is for menus or display process in design operation description language. A *PrimitiveTask* is a primitive design task which related to the most basic behavior taken by designers in design and have a syntax as follows.

$$\begin{array}{ll} PrimitiveTask & ::= \langle Assumption \rangle \mid \langle Demand \rangle \\ Assumption & ::= \langle Input \rangle \mid \langle Selection \rangle \\ Demand & ::= \langle Calculation \rangle \mid \langle Display \rangle \end{array}$$

where, the *Assumption* are concerned with the generation activities of data input, item selection, setting attribute domain, etc., the *Demand* are concerned in the activities of designers' demands to the system, such as the demand for design calculation, display for design solutions distribution or drawing graphics, etc. In these decision-making activities, designers perform judgements by their experience or knowledge.

5. SYSTEM CONSTRUCTION

5.1 Strategies for System Construction

We think the goal of knowledge based design support systems are not to imitate the problem-solving capabilities of human designers in design process, but is to provide a framework which is perceptible for designers and is competent for supporting the design process. In the other word, it should not be developed as such systems that aim to take the place of domain experts, like traditional expert systems, but should be the semi-automatic systems which can be made full use of the capabilities of computer systems as well as that of designers. that is, not only it needs to take care of the comprehension of designers on design knowledge and experience, especially the knowledge which is difficult to be formalized and represented in computer systems, but also to make good use of the capabilities of computer systems in iterative design calculation and data process. Based on this, we take some strategies in system construction which are described as follows.

Control of Design Process In order to control the design processes easily for designers, A design control panel is needed in system which can give a global description on design process, design objects as well as to support some basic behaviors such as editing, data input, solution select, vision confirmation about design objects, etc. With the control panel, designers can concentrate their attention on the design activities and pay few attention on data management. In addition, it also help the recruit designers in comprehension of the design process produced by experts. We present a tree-structure for design process control corresponding to integration of design process and assembly structure of design object. Figure 2 shows an example of vessel design problem.

where, in Figure2, a *QP* is a settled problem which can be model by quantitative approach.

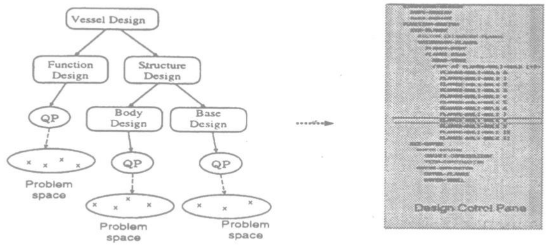


Fig. 2 **Control of design process**

Cell Process for Design Calculation Evaluation on settled design problems to find design solutions is done by full-quantitative approach in system like constraints satisfaction problem. The calculation problems are represented by generate-and-test calculation model or the other methods. A cell process is provided to model and represent the assumption, calculation and evaluation activities taken by designers in design calculation. By operation of cell process, the related calculation can be done automatically when calculating conditions are satisfied. In local design calculation, system provides the ability to search and generate numbers of design solutions systematically as well as to quote the related information automatically from database like catalog. In addition, the description about design solutions is in a suitable form in order to be selected easily, some pop-up menus, tables and graph are provided. Figure 3 shows a cell process for primitive design calculation problems.

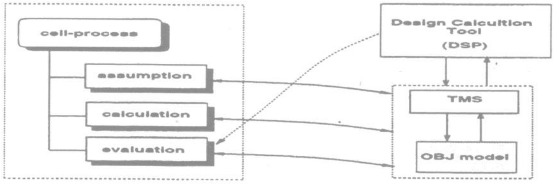


Fig. 3 **Cell process for design calculation problem**

where, the assumption is concerned about the data generation activities, such as data input, the calculation is related to design object to by the demand message from designers, and evaluation is for analysis and judgment of designers.

Design Object with Truth Maintenance Mechanism There are dependency relations between design attributes which maintain the integration of design objects. In design process, designers usually operate design object model arbitrarily to change the value of certain design attribute for getting new solutions. This includes editing design object attributes at certain design state, and perform calculation at certain point. This may destroy the integration indirectly with the progress of design process.

Truth maintenance mechanism is needed for design object to maintain the attribute dependency. It is concerned with the automatic correction of related property value.

Centralized Management of Design Data In general, design is a decision making process that generate the overall engineering information about a design object before it becomes a real product. For instance, draftings, design documentations, etc. Design information generated in each step are centrally arranged to relate with the design object model so as to let designers can pay less attention on data management in using various design tools at different design step.

5.2 SYSTEM ENVIRONMENT

A prototype system has been built to evaluate the ideas, concepts and the feasibility of system constructing approach. Figure 4 shows the system environment for designers to support the basic design activities.

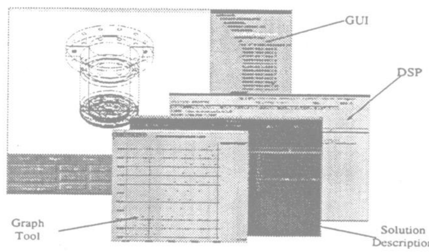


Fig. 4 **A support system environment**

The GUI is an interface for designers to control design process by process control panel and get vision confirmation by graphics display as well as to use the other support tools through it. The DSP is a work sheet for designers to support editing and processing design knowledge and performing design calculation. The solution description tool is for describing the distribution of design solutions or design results by tables, etc. The graph tool can give descriptions about the relationship among design attributes to support the decision-making for designers in the selection of optimum design solutions.

6. CONCLUSION

In this paper, we describe the construction of a design support system to support the design activities in the basic design phase of power plant heat-exchanger. The experimental study on several design problems such as pressure vessel design problem indicates that the method is capable of modeling design process and the arranged system environment is fit for the features of design activities in the basic design phase. Additional work is needed to test the approach in the actual design work of power plant heat-exchanger to get overall evaluation.

REFERENCES

[1] Isao Nagasawa, Masatoshi Ito: "Attribute Oriented Modeling", Simulation, Vol.10, No.2, pp110-118,1991.

[2] Hideyuki Yamaguchi, Zhihua Zhang, Isao Nagasawa, Masanobu Umeda: "Standardization of Design sequences of Heat-exchanger for Power Plant", Proceedings of ICMA'97, HK, Apr. 1997.

[3] Masanobu Umeda, Isao Nagasawa, and Tatsuji Higuchi: "The Elements of Programming Style in Design Calculations", Proceedings of the Nine International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems, June 4-7, 1996.

[4] Toyohiko Hirota, Masaaki Hashimoto, and Isao Nagasawa: "A discussion on Conceptual Model Description Language Specific for an Application Domain", Transactions of Information Processing Society of Japan, 36(5), pp.1151-1162, May 1995.

MODELLING THE HUMAN BODY FOR ERGONOMIC CAD

K.Case[1], J.M.Porter[2] D.Xiao[1] & B.S.Acar[3]
[1] Department of Manufacturing Engineering
[2] Department of Design and Technology
[3] Department of Computer Studies
Loughborough University, UK

ABSTRACT

A recently completed Brite-Euram (European Community) research project was concerned with life-cycle aspects of car seating with Loughborough University being responsible for driver comfort assessment. This was achieved by road and laboratory trials, with the results to be incorporated within the SAMMIE computer-aided ergonomic design system. Driver comfort is in part determined by seat pressure distributions which lead to deformation of the human flesh and the seat and result in uncertainty in the position of important design locations such as the driver's eyepoint. Accommodation of these effects requires a realistic representation of the human body using surface rather than solid representations. Hence a shadow scanning technique was used to capture human body shape which was processed into the DUCT surface modelling system and via IGES files into SAMMIE. Finite element techniques were then used to predict deformations at the seat/driver interface. Having established an anthropometrically correct representation of body shape, current research is aimed at improving the kinematic and analytic capabilities of the human model by introducing a multi-segment spine that can respond to external and internal loadings. This spine model is intended for use in the evaluation of human working postures (such as car driving) where, although the loadings might be viewed as well within human capabilities, previous studies have shown that back pain or damage might result. The model described is based on an arch representation rather than the pin-jointed rigid link systems which are perhaps more usual, but which have been shown to be deficient in several respects.

KEYWORDS

Ergonomic Design, Human Modelling, Spine Modelling

1. INTRODUCTION

The evaluation of the ergonomic aspects of design by the use of computer-aided design techniques is a well-established methodology and many computer systems are available [1]. SAMMIE, System for Aiding Man-Machine Interaction Evaluation [2] is a long-established and typical example that has been used in a wide variety of applications [3] and forms the basis of the work described here. Human modellers are frequently similar to more general kinematic modellers used for the simulation and evaluation of mechanisms such as industrial robots. Thus the major articulation points of the body are represented by pin-joints constrained to maintain motion within the ranges of human joint extensions. The joints are connected by rigid links as an approximation to the long bones of the body, and these are enfleshed to give a visual representation of human shape. Figure 1 shows a typical example of SAMMIE used in the ergonomic evaluation of a tractor and illustrates the symbolic nature of the flesh representation. Anthropometric control through manipulation of the joint-to-joint dimensions and flesh shape and size is provided so that the model can be made representative of the product user population. The complexity and variability of the human body has dictated that an approximate representation of shape is used (by the use of primitives in a boundary representation solid modelling scheme). This results in a symbolic shape, which although useful for many applications, is inadequate where the human body is in close contact with the product being designed (as in the case of a car seat). A second limitation arises from the complexities of the human skeletal link structure with its many degrees of freedom and the resulting difficulties in driving the model to predict realistic postures. Human models are often limited to twenty or so joints/links with the major area of approximation being in modelling the spine as only one or two links. Again this is adequate for many applications, but is not sufficiently precise elsewhere (such as car driving, difficult postures in assembly or machining, load carrying, etc). A final concern with existing modellers is their relative weakness in assessing loads on the body. This is particularly important for the spinal column as it serves as a major load–carrying member and is susceptible to long–term damage with even relatively light loading. These three limitations form the basis of current research and progress to date is described in this paper.

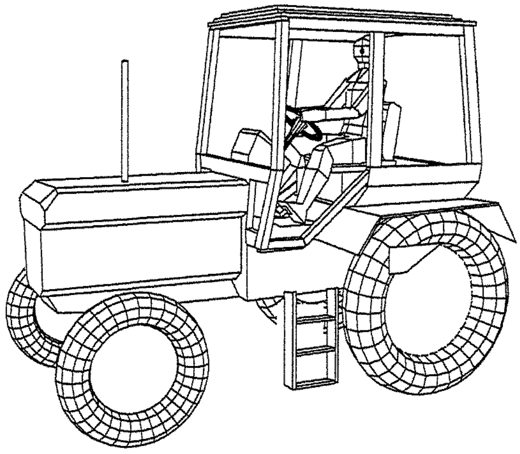


Figure 1. Evaluation of a Tractor using SAMMIE

2. MODELLING OF BODY SHAPE

.The variety in human body shape and dimension makes it particularly difficulty to determine initial shape representations and to subsequently manipulate these to represent individuals or general populations. A shadow scanning technique (LASS, Loughborough Anthropometric Shadow Scanner) developed by the HUMAG (Human Measurement and Growth) Research Group in the Human Sciences Department of Loughborough University was used to capture shape information. The entire human body can be scanned in a matter of minutes and ordered coordinate information provided. This information has been processed for input into the DUCT surface modeller where some data reduction can take place before being output in the form of IGES B–Spline surfaces. These surfaces are then processed into a quadrilateral mesh representation that can be handled by the PHIGS functionality implemented within SAMMIE.

2.1 Scanning Body Data

The Loughborough Anthropometric Shadow Scanner (LASS) [4,5] has been used to capture human body shape in the form of three–dimensional coordinates describing the surface. The subject stands on a platform which rotates through 360 degrees in steps of approximately 2.3 degrees in about 3 minutes. Strips of light are projected onto the body and recorded by television cameras. This produces a 'raw' binary data file typically containing some 64,000 coordinates. This data is processed both to remove 'noise' created by stray reflections and to extrapolate for regions that are obscured during the scanning process (e.g. the part of the torso hidden by the arms).

2.2 Surface Modelling

Coordinate data from the body scanner is used to create a surface model in proprietary computer–aided design systems (both DUCT and Unigraphics have been used for this purpose). Using the full data from the scanner gives a very precise representation of human shape as can be seen in figure 2(a) which shows a full resolution plot in both its rendered and wireframe forms. In this case the model consists of 36 horizontal sections each of which is defined by 30 three–dimensional coordinate points.

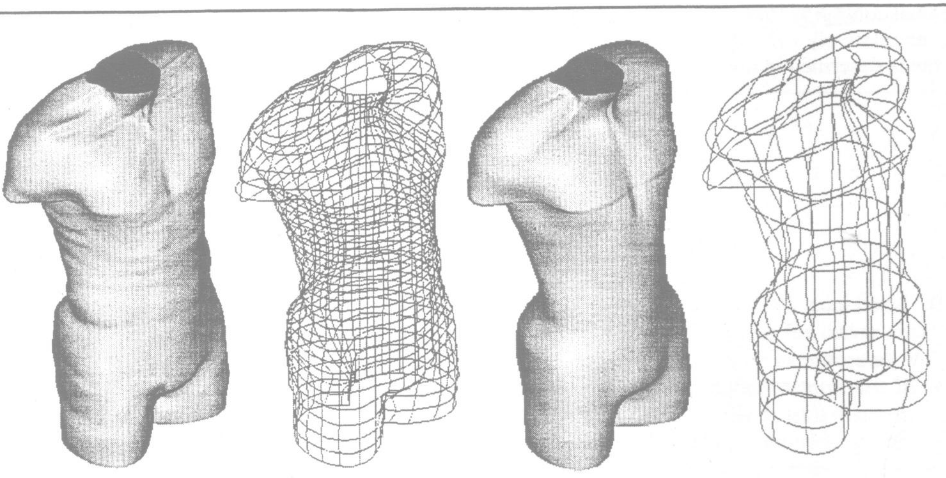


Figure 2(a). Male torso
(36 sections, 30 points per section)

Figure 2(b). Male torso
(11 sections, 16 points per section)

The information can be reduced while maintaining the anthropometric shape integrity. A simple reduction of the data is achieved by the removal of appropriate horizontal sections and alternate points on each section. Figure 2(b) illustrates that these simple techniques are effective in achieving an eighty–five percent data reduction in data. In practice however the surface modeller is used to convert the coordinate information into surface representations where control over the level of detail in the model can be controlled by the normal means available in such modellers.

2.3 Anthropometric Modelling

Merely capturing images of particular subjects and using this information to generate a model does not meet the requirements of an anthropometric model where there is a need to be able to manipulate the model such that it represents a user population. i.e. algorithms based on large scale surveys are required so that anthropometrically correct large/small, fat/thin, etc models can be synthesized and used in design situations. This is the subject of current research proposals, but we are helped in this by knowledge of the relationship between scan sections and anthropometric landmarks. The scan data also needs to be modified to allow us to generate a model that has joint articulation. This requires some modification of the surface models in the regions where limbs connect with each other and the torso. Figure 3 illustrates the current status of this work with an 'armless' figure (we have yet to obtain arm scans). The figure also illustrates the particular application of car seat evaluation where composite car seats have been constructed from parametrically defined elements such as seatpan, backrest, headrest, side supports

2.4 Transfer to SAMMIE

SAMMIE is a boundary representation solid modelling system and its graphics is handled by a commercial implementation of the PHIGS international standard [6]. The transfer from DUCT to SAMMIE is achieved by an IGES file containing B–Splines that are readily convertible into a mesh representation which reduces the visual realism of the image but is suited to the next stage of the work which is to use finite element methods to investigate the compression between driver and seat.

2.5 Analysis of driver/seat interface

The creation of more realistic body shapes fulfils an overall objective of improving the performance of the SAMMIE system, but for our Brite contract we wished to investigate and produce algorithms for the way in which the flesh and car seat deform when in contact. The Patran finite element package was used for this using the geometric knowledge contained in the models described above. Information on the compressibility of the seats was available from earlier experimentation and, although the compressibility of human flesh is less easily determined, the HUMAG research group had determined these characteristics whilst creating physical manikins. Our own experimental work has determined pressure contours for a variety of driver/seat configurations and these were used in the validation of our modelling work.

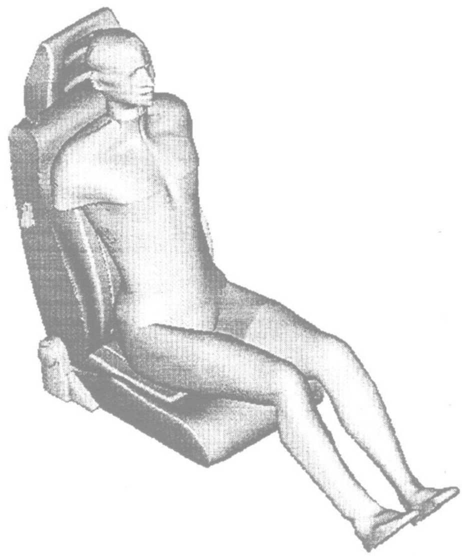


Figure 3. Human Model combined with parametrically defined model of a car seat

The investigation of deformations at the seat/human body interface had the objectives of determining variability in location of important design points (principally the eyepoint and the H–point) as drivers of different size and weight sit on seats with variable seat cushion compressibility, and the production of pressure distribution maps.

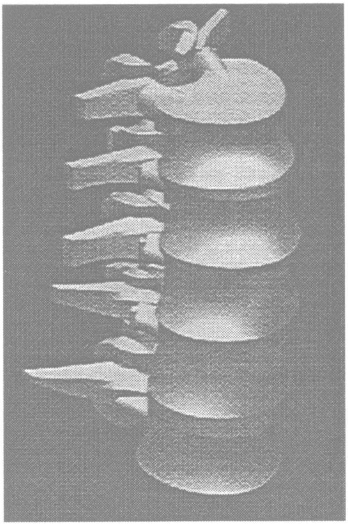
The pressure distribution maps were intended as a predictor of seating comfort – the hypothesis being that pressure distributions could be related to comfort and that this could be confirmed by our laboratory experiments. In the event this relationship proved to be difficult to find, so this aspect was not pursued. Subjective comparison of the experimentally and analytically obtained pressure maps has however provided us with a valuable confirmation of the validity of the analytical work.

Car seat evaluation is simply one example of an area where better representations of body shape are required, but there are many others. Any product that comes into direct contact with the body is a potential area of application of the technique – clothing, helmets, gloves and facemasks being typical examples. In other applications visual realism of the human figure itself is of importance. This is reflected in current research proposals to investigate and provide for the needs of virtual reality systems in this respect.

3. SPINE MODELLING

In addition to improving exterior shape representations, research is being undertaken that is aimed at developing parameterised spine models of sufficient geometric and kinematic complexity to provide a basis for analysing posture and loading. Eventual inclusion of this facility within the SAMMIE system will allow the evaluation of many of the complex comfort issues associated with a wide variety of human workplaces.

The creation of a geometric model of the spine is an essential precursor to the mathematical modelling work which is aimed at analysing loading conditions to predict discomfort or spinal damage. The geometric model [7] (produced using Unigraphics and illustrated in figure 4) describes the geometry of the spinal components (vertebrae and discs) and how these are spatially arranged (relative locations and orientations) to form the overall spine structure. A parametrically driven model is required to accommodate anthropometric variability between individual people.



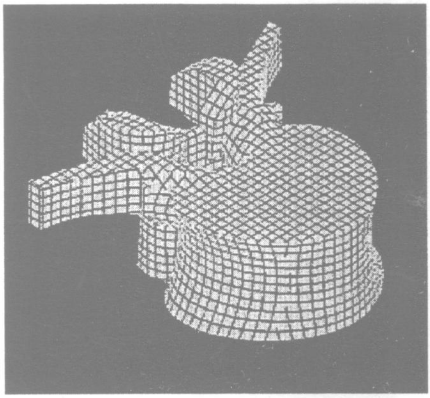


Figure 4. Model of a single Vertebra (above), and the assembled Lumbar Section of the spine (left)

The geometric model provides a starting point for the mathematical modelling of the spine. Study of the literature [8, 9] has established the range of methods available and in particular [10] provides a comparison between the conventional lever models and the arch model that we believe to be more appropriate for our work. The arch model draws an analogy between the human spine and masonry arches and it has been shown that in some circumstances it provides a more appropriate measure of internal loading at the vertebrae [11]. The purpose of our mathematical model is to predict the spinal component loads under fixed spine curvature conditions. Total spine loading occurs in three principal ways: (i) transmission of upper body forces due to gravity and external loads – this is represented as a force and torque vector applied to the cervical/thoracic spine; (ii) lumbar and thoracic gravity forces dependent on trunk size, shape and posture; and (iii) reaction forces on the spine due to muscles, ligament activity and intra–abdominal pressure.

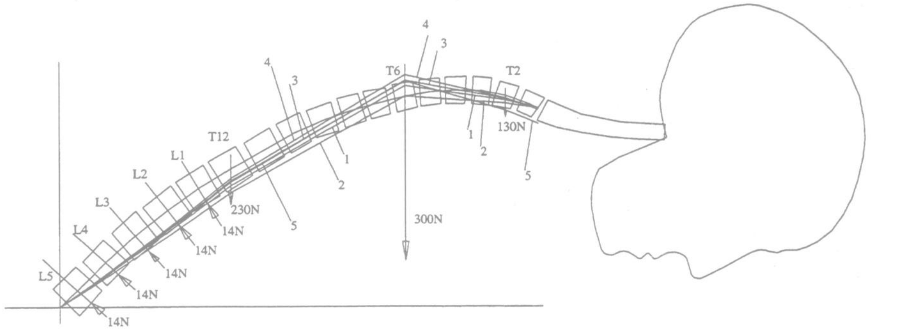


Figure 5. Spine Model showing load system and thrust line (stooped posture)

Prototype versions of the model have been constructed using techniques identified in the literature survey and combines aspects of both the lever and arch models. The ADAMS mechanism modeller is used to construct a hierarchical link model and to load it in the fashion indicated in figure 5. The first two loading conditions (external loads and trunk gravity forces) are those found in conventional lever models, while the arch model accommodates important aspects such as reaction forces due to muscular activity. This problem is statically indeterminate under these loading conditions and stability is determined by the application of well-known theories of the structural behaviour of arches [12]. Optimisation techniques are used to ensure that the thrust line is contained within the volume of the spine. The primary output of the model is the loading conditions at each of the vertebrae and discs which provide an indicator of the relative suitabilities of different postures, either by comparison with experimentally determined values or by further mathematical modelling using finite element methods. These aspects have not yet been tackled and constitute a significant part of our future work.

4. CONCLUSIONS

The work carried out to date and reported here has demonstrated the viability of the approach aimed at improving the capabilities of computer human models that are used for workplace design and evaluation. The more realistic flesh shape will not only improve visualisation, but will also open up many opportunities for a wider application of human modelling techniques. In particular, we are interested in products that have a direct interaction with the user (the car seat being a good example) and where there is potential for body and product shape to deform under normal loading conditions. The spine model with its considerably increased number of links will provide significantly improved posture representation and provides the basis upon which to evaluate the suitability of loading conditions and their relationships to potential discomfort and back injury.

5. ACKNOWLEDGEMENTS

The human shape modelling research described in this paper was a part of 'Advanced Technologies for Automobile Seat Evaluation and Design (SED)', a BRITE-EURAM Project (BRE2.CT92.0246) carried out in collaboration with Fiat, Courtaulds, Southampton University and Berlin University. The authors would like to gratefully acknowledge the invaluable support provided by Professor Peter Jones and the staff of the HUMAG Research Group in providing body scan data. The spine modelling research is funded by the Engineering and Physical Science Research Council (EPSRC) under grant GR/K58241, 'Spine Modelling: A Tool for Safe to Use Equipment Design'.

6. REFERENCES

1. Porter, J M, Freer, M T, Case, K & Bonney, M C. "Human Computer Modelling". In 'Evaluation of Human Work. A Practical Ergonomics Methodology'. 2nd Edition. Edited by J R Wilson & E N Corlett, Taylor & Francis, London. *In press.*
2. Porter, J M, Case, K & Freer, M T. "SAMMIE: Computer Aided Ergonomics". Co-Design, Vol 7-8, pp68-75, September 1996. ISSN 1335-302X, Published by Co-Design Partnership.
3. Porter, J M, Case, K and Freer, M T. "Recent Applications of the SAMMIE System". Proceedings of the International Ergonomics Association Conference, Tampere, Finland, July 1997.
4. Brooke-Wavell, K., Jones, P.R.M. & West, G.M. "Reliability and repeatability of 3-D body scanner (LASS) measurements compared to anthropometry", European Journal of Applied Physiology and Occupational Physiology, 68, pp514-518, 1994.
5. Jones, P.R.M., West, G.M., Harris, D.H. & Read, J.B. "The Loughborough anthropometric shadow scanner", Endeavour, New Series, Vol 13, No 4, pp162-168, 1989.
6. Gaskins, T. "PHIGS Programming Manual". O'Reilly & Associates, Sebastopol, California, USA, 1992.
7. Stepney, D., Acar, B.S., Case, K. & Porter, J.M. "Computer Aided Parametric Solid Modelling of the Spine", Proceedings of the Eurographics Conference, Imperial College, London, 1996.
8. Xiao, D.C. "Mathematical Modelling of the Human Spine – a literature review". Internal Report, Loughborough University, May 1996.
9. Xiao, D.C. "A review of Finite Element Modelling of the Human Spine". Internal Report, Loughborough University, May 1996.
10. Grilli, S L. "Towards a Computer-based Model of the Human Spine", P Department of Computer Studies, Loughborough University, December 1995.
11. Aspden, R.M. 'A New Mathematical Model: The Spine as and Arch', Spine, Vol 14, Part 3, pp 266-275, 1989.
12. Heyman, J. 'The Masonry Arch', Ellis Horwood Series in Engineering Science, Ellis Horwood Ltd, Chichester, UK, 1982.

A SIMULATION MODEL FOR IC ENGINE TURBOCHARGER-EXHAUST MANIFOLD DESIGN

Jingbin Yang and Thomas Campbell
Department of Energy and Environmental Technology
Glasgow Caledonian University
Glasgow, UK

ABSTRACT

Turbocharged internal combustion (IC) engine design and development is a very complicated process involving both an accurate matching of the turbocharger(s) and a correct design of the exhaust manifold. This paper describes the modelling and development of a computer simulation program for the engine turbocharger-exhaust manifold design. The gas flow in the exhaust manifold pipes is considered as non-homentropic, unsteady flow, and the modelling of the turbocharger compressor and turbine is extended to such an extent that even computational analysis of multi-turbocharger configurations, with complex arrangement either, in series, or in parallel, can be carried out. The program has been validated and used for the optimisational analysis of the turbocharger-exhaust manifold design for a turbocharged, six-cylinder, diesel engine. Some simulation results on changing turbocharger location and varying exhaust manifold cross sectional area are also reported.

KEYWORDS

Engine Design, Engine Modelling, Computer Simulation, Turbocharger and Turbocharging, Mathematical Analysis

1. INTRODUCTION

IC engine design and development is a very complicated process involving not only the dynamic structure analysis of various engine parts but also the detailed thermodynamic and gas dynamic analysis of the gas flow through the engine. For a turbocharged engine, an accurate matching of the turbocharger(s) and a correct design of the exhaust manifold is critical, as the mis-matching of the turbocharger(s) or improper design of the exhaust manifold can greatly affect the combustion processes of the engine, especially of multi-cylinder engines, resulting in a deteriorating of the overall engine performance. Nowadays, the thermodynamic and gas dynamic analysis has been overwhelmingly used in the turbocharger matching and the optimisation of the exhaust manifold design by means of specific computer simulation programs [1&2]. Most of the programs are developed from two fundamental modelling techniques, i.e. the method of characteristic (MOC) and the filling and emptying (F&E) method. These simulation programs have the advantage of being reliable and using less computer memory, therefore, they are widely used in both industry and academia for engine manifold design and turbocharger matching.

A common feature in these simulation programs is that one dimensional, non-homentropic analysis for unsteady flow in pipes is applied, and the turbocharger is only considered with simple matching applications. This means that only the turbine inlet and the compressor outlet of the turbocharger are associated with the above analysis. The turbine outlet and the compressor inlet are treated as open to atmosphere with a back pressure being added to the turbine outlet. This feature limits the simulation capacity of the programs, especially when multi-turbochargers are used and matched in different configurations, such as turbochargers being connected in series. To meet the requirement for current development of various turbocharging and supercharging systems and their matching techniques [3&4], the modelling of the turbocharger compressor and turbine has to be considered with optional inlet and outlet conditions.

The performance of a turbocharged engine is greatly affected by the exhaust manifold-turbocharger system. With a correctly matched turbocharger, the system is determined by the configuration of the exhaust manifold, such as the volume/size of the manifold. The systems can be classified as constant pressure or pulse turbocharging systems [5]. The former aims to achieve a steady turbine operation condition thus requiring a large manifold volume, whilst in the latter pressure fluctuations are taken account of, however, the size of the exhaust pipes and the way in which the turbocharger and the pipes are connected becomes important.

2. ENGINE MODELLING

The engine was divided into several parts and each part was modelled separately. The inlet manifold and the cylinders were treated as thermodynamic control volumes which were linked through the inlet valves. For each control volume, the transfers of mass, heat and work were calculated by evaluating the appropriate mass and energy conservation equations, and the rate of equivalence change equation on a crank angle basis.

$$\frac{dm}{d\theta} = \Sigma\left(\frac{dm(i)}{d\theta}\right)_j = \Sigma\left(\frac{dm(i)}{d\theta}\right)_{in} - \Sigma\left(\frac{dm(i)}{d\theta}\right)_{out} + \Sigma\left(\frac{dm_f}{d\theta}\right) \quad \text{Mass conservation} \quad (1)$$

$$\frac{dT(i)}{d\theta} = \left\{\left[\sum_{sf}\frac{dQ_{sf}(i)}{d\theta} + \sum_{in}\left(\frac{dH_0(i)}{d\theta}\right)_{in} - \sum_{out}\left(\frac{dH_0(i)}{d\theta}\right)_{out} + \frac{dm_f}{d\theta} - u(i)\frac{dm(i)}{d\theta}\right]\frac{1}{m(i)} - \frac{RT(i)}{V(i)}\frac{dV(i)}{d\theta} - \frac{\partial u(i)}{\partial F(i)}\frac{\partial F(i)}{\partial\theta}\right\} / \frac{\partial u(i)}{\partial T(i)} \quad \text{Energy conservation} \quad (2)$$

$$\frac{dF(i)}{d\theta} = \frac{F1(i)}{m(i)}\left(\frac{F1(i)}{f_{sto}}\frac{dm_{fb}(i)}{d\theta} - F(i)\frac{dm(i)}{d\theta}\right) \quad \text{Rate of equivalence change} \quad (3)$$

Note: *equivalence (F) may be defined as the product of actual to stoichiometric fuel-air ratio and,* $F1(i)=1+F(i)f_{sto}$

The combustion process occurring in the cylinders consisted of four stages, these being the intake, combustion, blowdown and scavenging stage. The thermodynamic process at each stage was different due to the mass transfer through the inlet and exhaust valve and the burning of the fuel. A previously developed single-zone model [6] was used to evaluate the fuel burning rate during the combustion process. The mass, heat and work were then determined by analysing the related conservation equations. A similar procedure was used for the inlet manifold, with the heat transfer being omitted. Mass flow through the inlet valve which links the manifold and cylinders was described by a constant pressure model, i.e. the valve throat pressure is assumed to be the same as the downstream pressure.

$$\frac{dm}{d\theta} = C_d F_{th} P_{up}\frac{30}{\pi N}\sqrt{\left\{\left(\frac{2\gamma}{\gamma-1}\right)\frac{1}{RT_{up}}\left[\left(\frac{P_{th}}{P_{up}}\right)^{2/\gamma} - \left(\frac{P_{th}}{P_{up}}\right)^{(\gamma+1)/\gamma}\right]\right\}} \quad \text{For sub-sonic flow} \quad (4)$$

$$\frac{dm}{d\theta} = C_d F_{th} P_{up}\frac{30}{\pi N}\sqrt{\left\{\frac{1}{RT_{th}}\left(\frac{2}{\gamma+1}\right)^{(\gamma+1)/(\gamma-1)}\right\}} \quad \text{For sonic flow} \quad (5)$$

Engine friction was directly related to the engine power output, in term of brake mean effective pressure (BMEP), and calculated by means of maximum cylinder pressure (P_{max}) and the piston moving velocity (V_{pis}).

$$\text{BMEP} = 0.114 + 0.037P_{max} + 0.125V_{pis} \quad (6)$$

The MOC method was adopted for modelling of the exhaust manifold to allow for the computation of the pressure fluctuations along the manifold. The conservation equations for compressible, unsteady flow were solved using this method. Three non-dimensional characteristics were obtained by consideration of the area change, friction, heat transfer and entropy gradient along each exhaust pipe, which was further divided into suitable meshes. They are the Riemann characteristic λ and β, and the path line characteristic A. These characteristics were calculated for each mesh point at each time step along the pipe. The pipe boundaries considered were, closed end, open end, exhaust valve, wastegate bypass valve, pipe junctions and turbocharger turbine. Turbocharger compressors were also considered to allow for all possible designs of turbocharger configurations. Detailed modelling processes for the pipe calculations and the valve and pipe junction boundaries have been described in Reference [7&8], only a brief discussion about the turbocharger modelling for different inlet and outlet conditions will be given here.

The various designs of turbocharging systems can be summarised, in terms of turbine/compressor inlet and outlet conditions, as volume-turbine/compressor-volume (VTV/VCV), volume-turbine/compressor-pipe (VTP/VCP), pipe-turbine/compressor-volume (PTV/PCV), and pipe-turbine/compressor-pipe (PTP/PCP) systems. In the modelling of these systems, the conventional performance map of both turbine and compressor, in the form of a non-dimensional mass flow rate $(\frac{m\sqrt{T_{in(0)}}}{P_{in(0)}})$, pressure ratio $(\frac{P_{in(out)}}{P_{out(in)}})$ and rotational speed $(\frac{N}{\sqrt{T_{in(0)}}})$ parameters, were inputted and held in a computerised matrix. At the VTV and VCV conditions, the mass flow rate and the instantaneous power of both devices were determined directly from the estimated turbocharger speed and the pressure ratio by interpolation of the performance matrix [9&10]. In other cases, the above performance characteristics had to be converted into appropriate Riemann variable forms. The conversion was based on the appropriate local Mach number at either inlet or outlet of the device, and the resulting equations are summarised as follows.

For PCV (PTV)

$$\lambda^{*}_{in1(3)/out1(3)} = (\frac{P_{1(3)}}{P_{2(4)}})^{\frac{\gamma-1}{2\gamma}} (1 \pm \frac{\gamma-1}{2} M_{1(3)}) \qquad \lambda^{*}_{out1(3)} = \lambda^{*}_{in1(3)} \left[\frac{1-\frac{\gamma-1}{2}M_{1(3)c}}{1+\frac{\gamma-1}{2}M_{1(3)c}} \right] \tag{7}$$

For VCP (VTP)

$$\lambda^{*}_{in2(4)/out2(4)} = (\frac{P_{2(4)}}{P_{r}})^{\frac{\gamma-1}{2\gamma}} (1 \pm \frac{\gamma-1}{2} M_{2(4)}) \qquad \lambda^{*}_{out2(out4)} = \lambda^{*}_{in2(in4)} \left[\frac{1-\frac{\gamma-1}{2}M_{2(4)c}}{1+\frac{\gamma-1}{2}M_{2(4)c}} \right] \tag{8}$$

For PCP (PTP)

$$\frac{\lambda^{*}_{out1(3)}}{\lambda^{*}_{in1(3)}} = \frac{1-C_{1(3)}G_{1(3)}}{1+C_{1(3)}G_{1(3)}} \qquad \frac{\lambda^{*}_{out2(4)}}{\lambda^{*}_{out1(3)}} = (\frac{P_{2(4)}}{P_{1(3)}})^{\frac{\gamma-1}{2\gamma}} \left[\frac{1+C_{2(4)}\frac{P_{1(3)}}{P_{2(4)}}\sqrt{\frac{T_{2(4)}}{T_{1(3)}}}G_{1(3)}}{1-C_{1(3)}G_{1(3)}} \right] \tag{9}$$

Where: $G_n = \frac{m\sqrt{T_n}}{P_n}$ $\qquad C_n = \frac{\gamma-1}{2}\sqrt{\frac{R}{\gamma}}\frac{1}{F_n}$ $\qquad$ (n=1...4)

3. VALIDATION OF THE ENGINE SIMULATION PROGRAM

The engine simulation program was written in standard FORTRAN 77, and was compiled on a Silicon Graphic workstation, which has a direct data access to the engine test bed data acquisition and operational control system. The calculation of the mass flow rate, temperature, pressure and other variants throughout the engine cycle was carried out through the solution of their derivatives with respect to crank angles. The fourth order Runge-Kutta integration method was used to compute the next step values, with the time step being set as one crank angle of engine revolution.

A Kelvin, four stroke, six-cylinder, turbocharged diesel engine was used for the validation of the above program. The output of main engine performance resulted from the simulation program was compared with the experimentation results and also with those obtained from a previously developed program using the F&E method [10]. As shown in Figure 1, a good agreement between the simulations and experimentation has been achieved for both programs, with the MOC method giving a more accurate result.

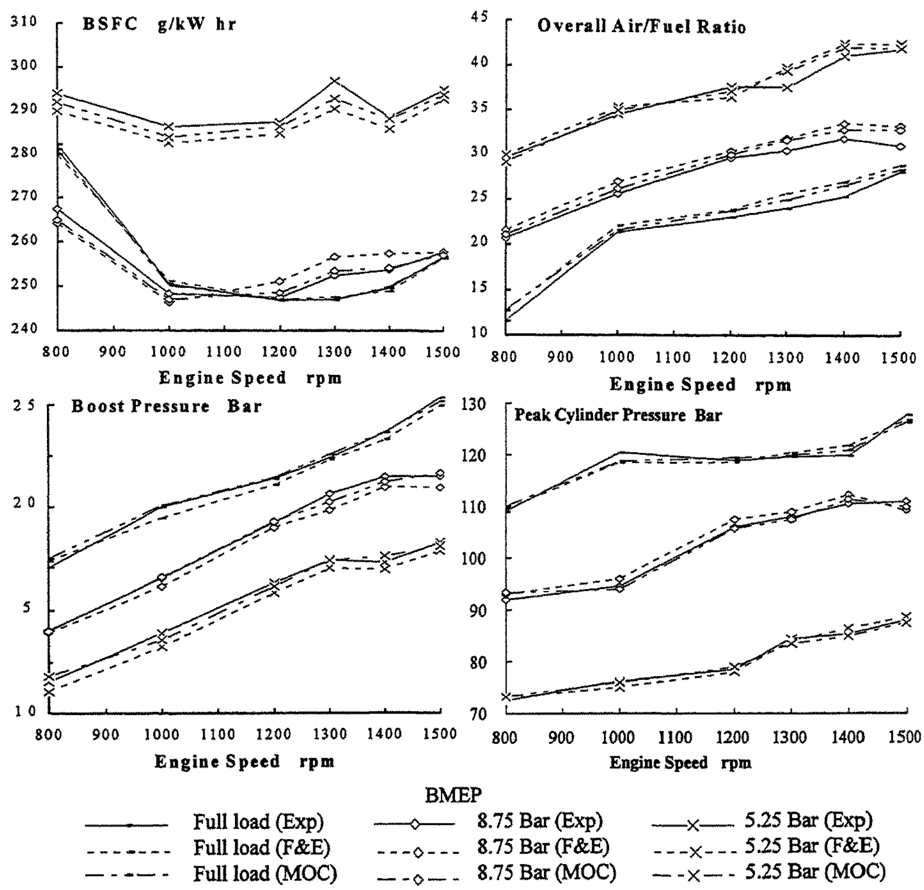


Figure 1: Comparison of engine performance between experimentation and simulations

4. DESIGN OPTIMISATION OF EXHAUST MANIFOLD-TURBOCHARGER SYSTEM

The above validated simulation program was then used to analyse and optimise the design of the exhaust manifold-turbocharger system for the above engine. The location of the turbocharger was first studied in consideration of the pressure wave effects on the turbine operation. The simulation results obtained from two possible positions of the turbocharger location, i.e. at the centre and at one end of the main exhaust pipe, were compared to those results from the current location. Table 1 shows the comparison of the brake specific fuel consumption (BSFC) of the engine, from which it can be seen that by changing the turbocharger location to one end of the main exhaust pipe the BSFC was reduced by 2 to 4 g/kWhr, whilst the BSFC difference between the current and centre locations were not distinct. As shown in Figure 2, this can be attributed to the improvement of the engine pressure gradient for better scavenge during the valve overlap period, caused by the improvement of the turbine inlet pressure.

Table 1 Predicted BSFC for different turbocharger locations and different pipe sizes

Speed	Load	Turbocharger location			Pipe Size	
rpm	BMEP	Current	Centre	End	8660 mm^2	9500 mm^2
1500	10.97	255.7	255.2	253.2	252.0	251.2
1500	8.77	254.2	252.9	250.9	249.6	248.9
1500	6.57	266.0	265.0	260.9	260.0	258.6
1300	11.78	250.2	249.6	246.8	245.5	244.4
1300	9.41	254.5	253.9	251.1	250.7	249.2
1300	7.06	267.7	265.8	262.2	261.0	259.7
1000	12.29	250.3	249.8	246.3	244.9	243.2
1000	9.74	248.8	247.3	244.7	243.1	241.6
1000	7.31	251.2	251.7	248.1	246.5	245.9
800	11.32	280.6	280.0	275.5	273.4	272.1
800	9.06	262.8	262.0	257.2	255.1	253.4
800	6.79	265.1	264.3	260.5	257.5	255.7

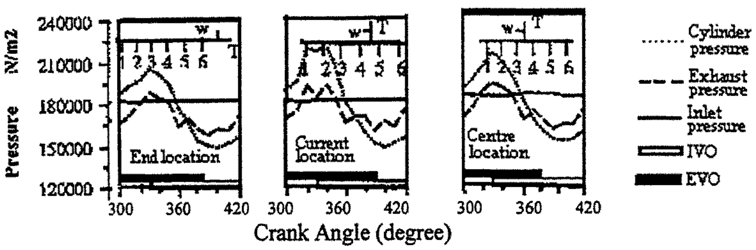


Figure 2 Engine pressure gradient with different turbocharger locations

Based on the above simulation results, an analysis on the enlargement of the main exhaust manifold pipe cross section area was carried out with the end turbocharger location. The cross sectional area was increased from the current size of 3454 mm^2 to 8660 mm^2 and 9500 mm^2 respectively. Although the pressure in the exhaust manifold was decreased and stabilised, the inlet pressure was also decreased, resulting in only a small decrease in engine BSFC (Table 1).

5. CONCLUSIONS

An IC engine simulation program has been developed to enhance the analysis of the design for exhaust manifold-turbocharger interactions. This was achieved by using the MOC method in the exhaust manifold pipe modelling, and extending the turbocharger modelling to all possible configurations. The modelling of the different parts of the engine, including the turbocharger turbine and compressor operating at different inlet and outlet conditions, has been briefly discussed.

The simulation program was validated using a Kelvin, six-cylinder, four stroke, turbocharged diesel engine. A good agreement between the simulation and experimentation results has been shown. The program was also used for the optimisation of the exhaust manifold-turbocharger design of the same engine, with the predicted results showing that a better engine performance could be achieved by changing the current turbocharger location to one end of the main exhaust pipe. Comparing the small improvement of the engine BSFC by increasing the exhaust pipe cross section area with the increase in the engine overall size, the enlargement of the exhaust pipe cross section area is not recommended.

The above developed engine simulation program has been greatly enhanced in its capacity for modelling and analysis, extending its application to alternative exhaust manifold turbocharging system designs.

6. NOMENCLATURE

C_d	coefficient of discharge	f	burnt Fuel/air ratio
F	equivalence ratio or cross sectional area (m^2)	H	specific enthalpy (*kJ/kg*)
i	control volume number	m	mass (*kg*)
M	Mach number	N	revolutionary speed (*rpm*)
P	pressure (N/m^2)	Q	heat transfer rate (*kW*)
R	ideal gas constant (*kJ/kg K*)	T	temperature (*K*), (turbine in Figures)
u	specific internal energy (*kJ/kg*) or gas velocity (*m/s*)		
V	volume (M^3)	W	wastegate in Figures
γ	ratio of specific heat (C_p/C_v)	λ^*	starred Riemann variable
θ	angle (°,rads)		

Subscript

c	choke condition	*fb*	fuel burnt for formation
j	control volume port	*o*	stagnation condition
for	formation	*r*	reference condition
sf	surface	*sto*	stoichiometric
th	valve throat	*up*	upstream condition
1	compressor inlet	*2*	compressor outlet
3	turbine inlet	*4*	turbine outlet

7. REFERENCES

1. Connor, W.A. and Swain, E., 'Extension of the filling and emptying engine performance simulation method to include gas dynamic effects', Proc. of IMechE: Turbocharging and turbochargers, C484/042, pp. 1-12, IMechE, 1994.
2. Hirst, P., 'A simple 'field proven' engine performance prediction program', Proc. of IMechE: Turbocharging and turbochargers, C484/032, pp. 21-37, IMechE, 1994.
3. Campbell, T. and Galbraith, J., 'Sequential turbocharging of a medium speed marine diesel engines', Proc. of IMechE: Turbocharging and turbochargers, C433/029, pp. 31-47, IMechE, 1991.
4. Abdel Rahman, A.A., Ibrahim, M.K. and Said, A.A., 'Improvement of part load performance of diesel engines operating at constant speed conditions', Jour. of Power & Energy, Proc. of IMechE Vol. 208, No 1, pp. 21-25, 1994.
5. Bulaty, T., 'The use of computer simulation to improve turbocharging of diesel engines', Jour. of ASME, Vol. 65, pp. 27-33, 1989.
6. Watson, N., Pilley, AD and Marzouk, K., 'A combustion correlation for diesel engine simulation', SAE Paper, 800029, 1980.
7. Yang, J., Investigation of wastegate turbocharging systems for medium speed diesel engines, PhD Thesis, GCU, 1994.
8. Benson, R.S., The thermodynamics and gas dynamics of internal-combustion engines, Vol. 1, Clarendon Press, Oxford, 1982.
9. Yang, J. and Campbell, T., 'Development of an IC engine simulation programme for multiple turbocharger configurations', SAE Paper, 970056, 1997.
10. Yang, J. and Campbell, T., 'Optimization of a wastegate turbocharged medium speed diesel engine', SAE Paper, 930195, 1993.

COMPUTER-AIDED MECHANISM DESIGN FOR CONSTRAINED FUNCTION GENERATION IN A ROBOTIC AUTOMATION CELL

Zhongming (Wilson) Liang
Purdue University Fort Wayne
Fort Wayne, Indiana, USA

ABSTRACT

This paper reports an effective computer-aided design of a mechanism that meets the function generation requirements and various constraints. The mechanism helps increase the automation level of the computer-integrated manufacturing cell in the laboratory.

KEYWORDS

Mechanism Design, Function Generation, Design Constraints

1. INTRODUCTION

A Spectralight CNC turning machine, an educational table-top CNC, is a component in a computer-integrated manufacturing cell in an automation laboratory of the department. For a robot's hand to reach the chuck for workpiece loading and unloading, the safety shield of the CNC was sometimes removed, which was not most appropriate from the safety point of view. A mechanism was desired automatically to open the shield for the robot hand and to close the field during machining processes. Various designs of the mechanism by the manufacturer of the CNC machine and by some universities were reviewed. Typically, they were four-bar linkages and had the following disadvantages: (1) a large space occupied by a simple slider-crank mechanism; (2) undesired kinematic performance such as a significant velocity of the safety shield when it closes; (3) unnecessarily large actuators such as pneumatic cylinders.

The author of this paper designed the mechanism using the following approach: (1) a six-bar mechanism was synthesized to meet the requirement of the safety shield motion range, the requirement of the correspondences between the actuator positions and the safety shield positions, the space requirement and the force requirement; (2) the motion characteristics of the mechanism were examined analytically and using computer simulation; (3) the stresses in key elements of the design were examined by using finite element analysis.

Design methods of four bar linkages without additional constraints are widely available in literature. Rarely available are design approaches of linkages of more than four links and with additional kinematic and dynamic constraints such as the one presented in this paper.

2. PROBLEM STATEMENT AND CONSTRAINTS

2.1 Problem Statement

Design a mechanism to rotate the safety shield of the Spectralight CNC machine between the 0-degree closed position and the 80-degree open position.

2.2 Design Constraints

- The mechanism should be located on the backside of the CNC machine and should occupy as little space as possible.
- The mechanism should be so designed that it will not block the operator's view of machining processes under the safety shield
- The available pneumatic actuator has a stroke of 4 inches and a piston diameter of 1.125 inches. With the air pressure being 50 psi, the maximum actuating force is 49.7 lbs.

- The velocities of the safety shield should be small at the ends of its opening motion and closing motion to minimize impact, vibrations, and noises.
- The plastic safety shield should not skew during motion.

3. DESIGN OF THE MECHANISM

3.1 Determine the Length of the Output Rocker of the Mechanism

The designed is a slider-rocker mechanism, where the piston of the pneumatic cylinder is the input link and the safety shield is the output link. Figure 1 shows the mechanism viewed from the right side of the CNC machine. In the figure, point P is the fixed pivot of the safety shield and D_1 and D_2 are the positions of the movable pivot in two extreme positions of the shield. The movable pivot is mounted on the safety shield and connects it to the rest of the mechanism. The length of the output link is the distance from the fixed pivot to the movable pivot of the safety shield.

From the point of the operator's view of the CNC processes under the safety shield, it is better to locate its movable pivot and the connecting links of the mechanism closer to the back of the CNC, which means a shorter length of PD. However, a shorter PD means a smaller moment arm for the force to rotate the safety shield and hence requires a larger actuating force.

Static force analysis indicates that under the condition of the maximum actuating force the minimum length of PD should be 5 inches.

When viewed from the top of the CNC machine, the mechanism is placed in the middle across the length of the safety shield so that it would not skew when it is lifted up.

3.2 Design Procedure

- Select the position of the centerline of the pneumatic actuator. Under the space constraint, the centerline is best set vertical and as close to the fixed pivot P of the safety shield as much as possible. See Fig. 1.

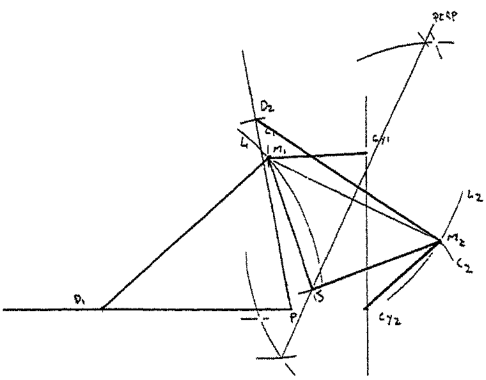


Fig. 1. Synthesis of the Mechanism

- On the centerline of the pneumatic actuator, select the two extreme positions of the pivot mounted to the upper end of the pneumatic piston. The positions are labeled as C_{y1} and C_{y2} in Fig. 1.
  - The distance between the two positions is chosen to be the length of the full stroke of the piston, which means that the piston moves for its whole stroke in both directions. The important advantage of this design choice is to satisfy the speed constraint of the design. Since the piston slows down at the end of each full stroke thanks to the cushion

design in the cylinder, the speeds of the safety shield are low at the ends of its opening motion and closing motion.

- The absolute positions of C_{y1} and C_{y2} are determined based on the space constraint: the height of the CNC machine base and the length of the pneumatic cylinder.

- Draw an arc L_1 of a selected radius R_1 and a center at D_1 and an arc C_1 of a selected radius R_2 and a center at C_{y1}. The two arcs intersect at point M_1.

- Draw an arc L_2 of the radius R_1 and a center at D_2 and an arc C_2 of the radius R_2 and a center at C_{y2}. The two arcs intersect at point M_2.

- Draw a line to connect point M_1 to point M_2. Draw a perpendicular bisector PERP of the line M_1M_2.

- Select a point S on the line PERP as the fixed pivot point in the middle of the six-bar mechanism.

In Figure 1, $PD_1M_1SC_{y1}$ is the closing configuration of the mechanism, in which P and S are the fixed pivot points, D_1, M_1 and C_{y1} are the movable pivot points, PD_1 is the safety shield, D_1M_1, M_1S and M_1C_{y1} are the links, and the vertical line through C_{y1} is the axis of the piston rod. $PD_2M_2SC_{y2}$ is the opening configuration of the mechanism.

Multiple mechanism models have been found in this stage. The best among them has been selected in the analysis phase, which will be discussed in the next section.

4. ANALYSIS OF THE MECHANISM

4.1 Motion Analysis

- Analysis with DesignView

DesignView, a ComputerVision CAD product for design applications, has been selected for motion analysis. The software, designed based on variational geometry, allows geometric constraints and equations to be attached to geometric elements. The software is used to review how the mechanism changes its configurations when positions of the pneumatic piston, dimension I in Figure 2, are varied. Using the dynamic data exchange (DDE) function of the Windows, the data of the piston positions and the safety shield positions are also sent to Microsoft Excel as shown in Figure 3 and Figure 4.

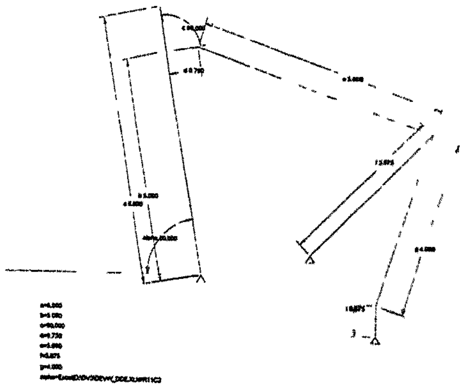
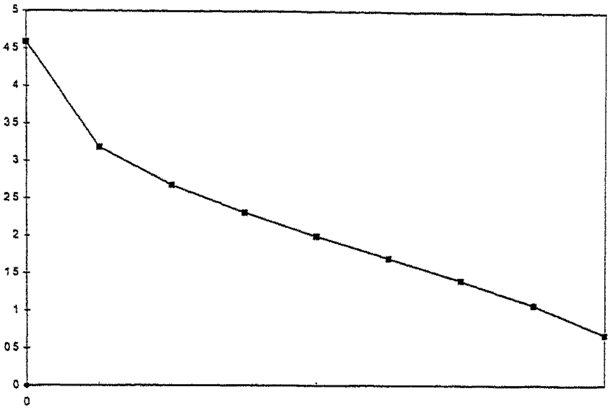


Fig. 2 Motion Analysis with DesignView

	I	J
6		
7	0	4 5876804
8	10	3 1833718
9	20	2 6764385
10	30	2 3075431
11	40	1.9940415
12	50	1 7000067
13	60	1.400555
14	70	1 0704096
15	80	0 6753006

I - Shield positions; J - Piston positions

Fig. 3 Data in Excel from Dynamic Link with DesignView



Horizontal coordinates - Shield positions; Vertical Coordinates - Piston positions

Fig. 4 Graph in Excel from Dynamic Link with DesignView

4.2 Velocity Analysis

- Velocity Analysis with Working Model

Working Model, a software product by Knowledge Revolution capable of complex simulation and analysis of dynamic mechanical systems, has been selected for velocity analysis. The analysis allows friction, weights, and moments of inertia to be taken into considerations in a simple manner, as shown in Figure 5.

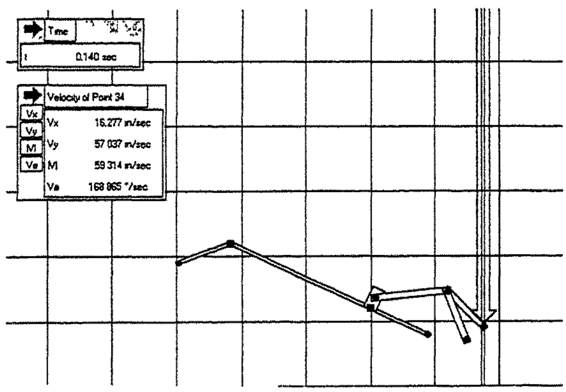


Fig. 5 Velocity Analysis in the Working Model

4.3 Force Analysis

- Analytical Force Analysis

Force analysis in connection with motion analysis of the mechanism is performed analytically with use of Microsoft Excel for digital computations and output formatting. The analytical formulas coded into Excel equations are shown in Figure 6. The graphical output in Figure 7 indicates that the maximum piston force required is less than 25 lb and thus the force constraint of the design is met. The graphical output in Figure 8 states that the angular velocity of the safety shield is small between 0 degree and 5 degrees and hence the velocity constraint is met.

Kinematic Analysis	
Shield Angle theta_f	0
Angle Theta_fe	11 31
Angle Theta_ab	8 531
Angle Theta_2	=180-B2 B3-B4
Link length r_1	2 55
Link length r_2	5 0559
Link length r_3	5 69
Link length r_4	3 875
Length d	=SQRT(B6^2+B7^2-2*B6*B7*COS(B5*PI()/180))
Angle Theta_d	=ATAN(B7*SIN(B5*PI()/180)/(B6-B7*COS(B5*PI()/180)))*180/PI()
(Theta_3 + Theta_d)	=ACOS((B8^2+B10^2 B9^2)/(2*B10*B8))*180/PI()
Angle Theta_3	=B12-B11
Angle Theta_b*	=B4+B3+B2+B13
Angle Theta_b	=B14+90-B4
Tangent(Theta_4)	=(B10*SIN(B11*PI()/180)+B8*SIN(B13*PI()/180))/(B10*COS(B11*PI()/180)-B8*COS(B13*PI()/180))
Angle Theta_4	=IF(ATAN(B16)>0 ATAN(B16)*180/PI() ATAN(B16)*180/PI()+180)
Angle Theta_c1	=180-B17-B13
Angle Theta_4*	=B17-B3
Distance e	1 5
Link length r_5	4
Angle Theta_5	=ACOS((B20+B9*COS(B19*PI()/180))/B21)*180/PI()
Angle Theta_c2	=B19 B22
Length g	1 75
Piston rod position l	=B24+B9*SIN(B19*PI()/180)-B21*SIN(B22*PI()/180)
Force Analysis with the weights of the links ne	
Shield weight (1)	2 94
Sh W (1) moment arm	=8 75*COS(B2*PI()/180)
Shield weight (2)	0 76
Sh W (2) moment arm	=19 08*COS((B2-5 25)*PI()/180)
Moment arm of F_bc	=B7*SIN(B14*PI()/180)
Force F_bc	=(B27*B28+B29*B30)/B31
Angle Theta_c1	101 671
Angle Theta_c2	48 065
Force F_cd	=B32*SIN(B18*PI()/180)/SIN(B23*PI()/180)
Piston force (lbs)	=B35*SIN(B22*PI()/180)+0 2*B35*COS(B22*PI()/180)

Fig. 6 Excel Equations for Force Analysis

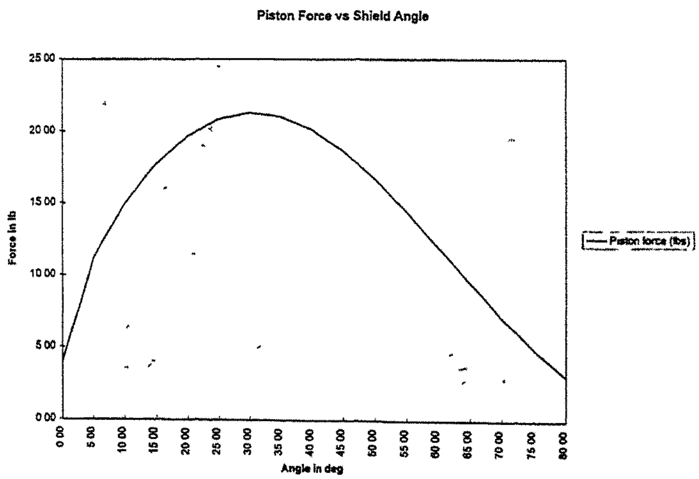


Fig. 7 Excel Graphical Output of Force Analysis

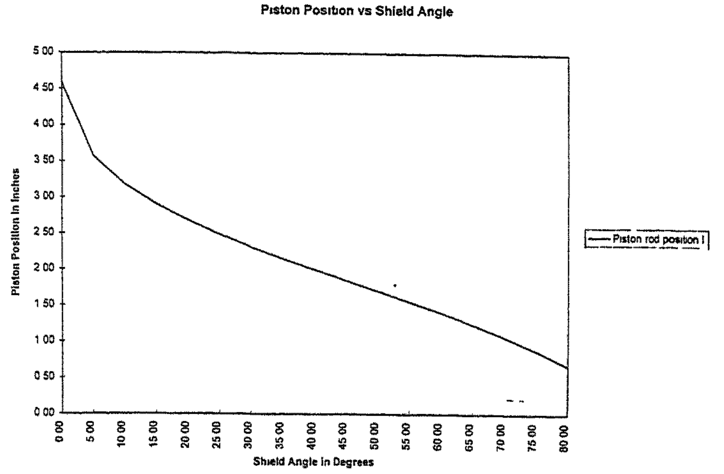


Fig. 8 Excel Graphical Output of Position and Velocity Analysis

4.4 Stress Analysis

- Finite Element Stress Analysis

Images 3D by Celestial Software has been selected for finite element analysis of the some elements of the design. The easy-to-use software permits efficient and effective analysis of the structure for stresses and elastic deformations.

5. CONCLUSIONS

An innovative kinematic design method has been presented in the paper. The resulting multi-link mechanism meets the basic requirements of the output link positions and the input-output functional relationship. It satisfies various design constraints of space, location, force, velocity, and elastic deformation. The design method has been developed based on basic principles of mechanism synthesis and has been analyzed in various aspects and by using analytical methods in combination with computer simulation. In comparison with similar devices available in labs of many other places, the mechanism built here is much more compact and much more energy-efficient.

It should be worth mention also that the student participating in the design project was very satisfied with his learning experience in this creative work. He wrote that "I have been given an opportunity that most students at IPFW[1] do not get.... I hope that future students and professors have chances to do the type of projects that Prof. Liang and I have been able to."

6. REFERENCES

1. Erdman, A.G. and Sandor, G.N., Mechanism Design: Analysis and Synthesis, Volume 1, third edition, Prentice-Hall, New Jersey, 1997.
2. Sandor, G.N. and Erdman, A.G., Advanced Mechanism Design: Analysis and Synthesis, Volume 2, Prentice-Hall, New Jersey, 1984.
3. Reddy, J.N., An Introduction to the Finite Element Method, second edition, McGraw-Hill, New York, 1993.
4. Chang, T.C., et al., Computer-Aided Manufacturing, Prentice-Hall, New Jersey, 1991.
5. Rembold, U., et al., Computer Integrated Manufacturing and Engineering, Addison-Wesley, New York, 1993.

[1] Indiana University- Purdue University Fort Wayne

A NEW CNC SYSTEM WITH SCULPTURED SURFACE INTERPOLATION CAPABILITY

Yanhong Zhou, Yong Zhan and Ji Zhou
School of Mechanical Science & Engineering
Huazhong University of Science & Technology
Wuhan, P R CHINA

ABSTRACT

A new CNC system with a real-time algorithm for surface interpolation has been developed based on 80486 industrial control computer. The system provided the ability to on line modify the processing parameters concerned with the structure of machine tools, the shape and dimension of cutters, the allowance for finish, etc., to calculate the path interval according to the given cusp height, and to do interference checking and handling in real time. The system can be applied to machine complicated surfaces such as compound surfaces and arbitrary pockets. The input information only consists of the surface geometry and boundary definitions as well as the processing parameters, and a high-level language similar to APT can be used for programming. Finishing surfaces has been more efficiently gained at lower cost. In this paper, the system's structure, functions and working principle are presented. The key techniques for the planning of machining path, the generation of real-time interference-free tool path and the feedrate control, etc., are given.

KEYWORDS

CNC, Sculptured Surface, Surface Machining, Interpolation

1. INTRODUCTION

Sculptured surface is an important feature in many engineering parts, such as turbine blades, automobile bodies, etc. At present, most CNC systems only offer the interpolation functions for straight line, circular arc and a few kinds of curves, so the complex curved surfaces have to be machined in a CNC system as a collection of fine linear or circular segments generated by an off-line programming procedure. It is imperative for the high quality surface machining to approximate the surface contour in extremely tiny steplength and dense sidestep, thus the part programs (NC codes) are very large, and it is inconvenient to generate, verify, operate and preserve such information. Moreover, once the part programs have been generated, the processing parameters including the tool dimension, the allowance for finish, etc., can not be adjusted in CNC systems, and the part programs are inefficient when the machining condition is changed. Furthermore, roughing and finishing have to be programmed respectively. Obviously it is also difficult for operators to bring their ability into full play.

The basic control of CNC is the control of motion. It is an effective way for economical and efficient surface finishing to enhance CNC's interpolation function for complete path. The real-time algorithm for surface direct interpolation used in this system makes, in this paper, CNC systems can machine engineering curves and surfaces directly. A high-level language similar to APT can be used for programming, and the processing parameters can be modified on line. It has been overcome the some problems finishing surfaces with the general CNC.

2. THE CONCEPTION OF SURFACE DIRECT INTERPOLATION

During the course of direct interpolation for machining surface with the CNC, a CNC system automatically generates the real-time toolpath for machining surface and controls the motion of the machine tool. The definition of direct interpolation for machining surface with the CNC can be expressed as: given machining information of the surface to be machined, boundary information, machining way and feeding mode, definition of cutter, allowance for finish, cusp height, feedrate, etc., a CNC system could in real time generate the machining path, interpolate the toolpath which meets the requirement of the feedrate and control the motion of coordinate axes to complete the surface machining according to the structure of the actual machine tool.

Let $r_s = r_s(u, v)$ be a curved surface, then the parametric equation of various machining path could be expressed as: $r_s(t) = r_s(u(t), v(t))$. In the five-axis machining shown in the Fig. 1, the tool path for cutting along $r_s(t)$ is given by:

$$\begin{cases} r_p(t) = r_s(\mathrm{t}) + r_\delta(\mathrm{t}) + r_c(\mathrm{t}) \\ \quad = r_s(\mathrm{t}) + (R_1 + b)\cdot n(\mathrm{t}) + (R_2 - R_1)\cdot\{n(\mathrm{t}) - [n(\mathrm{t})\cdot u(\mathrm{t})]\cdot u(\mathrm{t})\} / |n(\mathrm{t}) - [n(\mathrm{t})\cdot u(\mathrm{t})]\cdot u(\mathrm{t})| \\ u(\mathrm{t}) = \sin\alpha\cdot a + \cos\gamma\cdot n \pm (\sin^2\gamma - \cos^2\alpha)^{1/2}\cdot v \end{cases} \quad (1)$$

where R_1, R_2 are the parameters of the cutter, $n(t)$ and $a(t)$ are the unit normal vector and the tangent vector in the feeding direction at the cutter contact point respectively, $v(t) = n(t)\times a(t)$, (a, v, n) is the local coordinate system at the cutting point, tilt angle α and yaw angle γ are the control angles of the cutter axis in (a, v, n), $u(t)$ is the unit vector of the cutter axis, $r_\delta(t) = b\cdot n(t)$ is the vector for allowance compensation, b is the allowance for finish in the direction of the normal vector of the part surface, $r_c(t)$ is the offset vector of the cutter, $r_p(t)$ is the positional vector of the cutter location point p.

For the system shown in the figure 1, let the axis A be installed on the axis B, and the axis B and the axis A intersect at o_m, then the translation of the machine tool could be considered as the motion of point o_m. It can be expressed as: $r_m(t) = (x_m(t), y_m(t), z_m(t))$. Let $\varphi(t)$ and $\theta(t)$ be the rotational motions of the axis B and the axis A respectively, then the movement of coordinate axes can be given by

$$\begin{cases} r_m(t) = r_o + r_p(t) + l\cdot u(t) \\ M_B(\varphi(t))\cdot M_A(\theta(t))\cdot u(0) = u(t) \end{cases} \quad (2)$$

where r_o is the positional vector of the workpiece coordinate system in the machine coordinate system, $M_B(\varphi(t))$ and $M_A(\theta(t))$ are the matrixes of the cutter axis rotating around axis B and axis A respectively, $l\cdot u(t)$ is the vector of the cutter axis, l is the distance between the cutter location point p and the swinging center o_m.

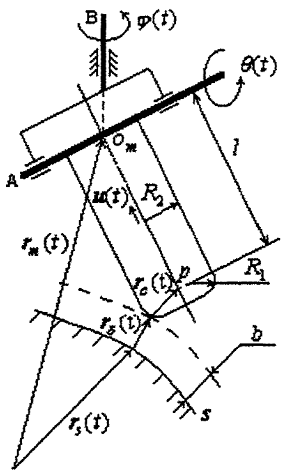


Fig. 1 : The calculating of toolpath for five-axis machining.

The digital sampling interpolation method based on time subdividing is adopted in most closed-up CNC system. During the course of the surface direct interpolation for CNC machining, the interpolating during the work period K is to calculate the axial moving increments $(\Delta r_m(t_k), \Delta\varphi(t_k), \Delta\theta(t_k))$ in the work period K+1 in real time, which should meet the requirement of the given feedrate. They can be expressed as:

$$\begin{cases} \Delta r_m(t_k) = r_m(t_{k+1}) - r_m(t_k) \\ \Delta\varphi(t_k) = \varphi(t_{k+1}) - \varphi(t_k) \\ \Delta\theta(t_k) = \theta(t_{k+1}) - \theta(t_k) \end{cases} \quad (3)$$

Given the starting point and endpoint information of $r_s(t) = r_s(u(t), v(t))$ and repeating the above calculation, the direct machining of the cutting path for could be completed. At the same time, according to the requirements including geometric data, boundary information, tool shape and dimension, machining way, cusp height, etc., The machining path (containing the starting point and endpoint information) and auxiliary approaching or retracting path can be generated , and then the surface machining can be completed by the above path machining functions.

3. THE FUNCTIONS, STRUCTURE AND KEY TECHNIQUES OF THE SYSTEM

3.1 The Functions of The CNC Direct Machining System

a. The types of surface which could be machined: general quadratic surfaces; surfaces generated by revolving a straight line, circular arc or spline curve; bicubic parametric surfaces; ruled surfaces defined by two parametric curves; compound surfaces or arbitrary pockets composed of the above surfaces.

b. The ability of toolpath process: simultaneous (3~5)-axis control; the cutting tools including ball end cutters, torus cutters and flat end cutters; detecting and handling interference in real time; the control of general tool direction in five-axis machining; the automatic adjusting of path interval according to the given cusp height; the machining ways including parametric, CC Cartesian, CL Cartesian and surface intersection machining; the compensation of normal machining allowance.

c. The ability of machining movement control: the control of all kinds of (3~5)-axis mechanisms and the compensation of structural errors; the acceleration and deceleration control in the index or linear rule in the direction of synthetic feedrate; automatic adjusting of feedrate to decrease the jerky motion and non-linear machining errors in the machining of surface with big curvature.

d. Other functions: the three-dimensional graphic verification of part surface and toolpath; the real-time graphic display of machine tool movement; the transformation of surface data, etc.

3.2 The Structure of Software

The software system's structure for surface direct machining in CNC system is shown in the Fig. 2, where the part program is deciphered and preprocessed in the decoder; The screen editing can be applied to on-line modifying processing parameters, the data of surfaces, the structure and parameters of the machine tool, etc. The generation of the axial movement command is adopted to transform the cutter location data into the data of axial movement according to the structure of the actual machine tool. The validity of toolpath can be checked by the kinetic simulation. The actual 3-D movement of the machine tool can be displayed on the screen in real time.

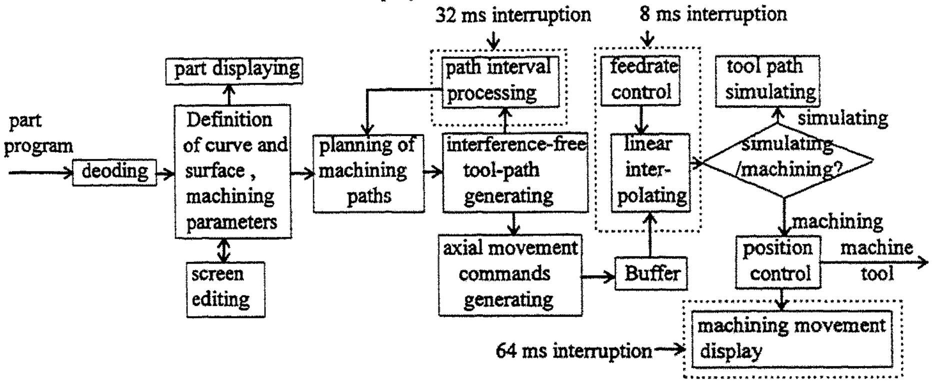


Fig. 2 : The system's structure of software for surface direct machining.

3.3 The Planning of Machining Path

The planning of machining path can in real time calculate the machining path and the auxiliary approaching and retracting path according to the defined machining way, the type of surface to be machined and the path interval. It is rather complex to plan the path for machining the area of complicated shape and the compound surface in high efficiency and reliability. The procedure for machining the area with multiple-island, in this system, is given by: [1]

a. the preprocessing of boundary curve. It is convenient for the subsequent processing to make the data of area and boundary definition transforming their coordinate values, discreting (to the free form curve) and arranging in the machining direction.

b. the processing of islands. The virtual boundary is used to integrate the boundary of area with that of islands so that the machining procedure of multiple-island area can be the same as area without islands.

c. the processing for interference-free path of the contour of boundary. The offset curves of the contour of boundary are generated, and the data of local or whole interferential areas (uncut areas) are removed and preserved for the convenience of the subsequent machining with a smaller cutter.

d. the separating of areas. The machining area which boundary is composed of the offset contours is separated into a collection of convex subareas in the machining direction. A subarea is the minimum machined unit.

e. the calculating for toolpath of machining subarea. The starting and end CL points of machining paths as well as the transitional sequence of CL points between machining paths are calculated.

f. the connecting for toolpath of machining subarea. The sequence of machining subareas is optimized in the rule of making vacant stroke minimum for improving machining efficiency.

3.4 The Generation of Real-time Interference-free Toolpath

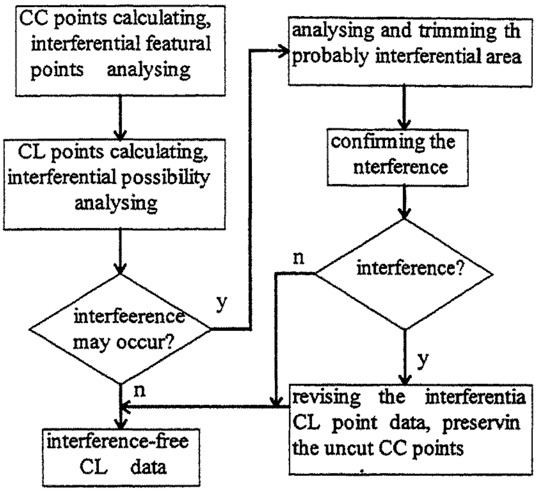


Fig. 3 : The Procedure for Generating Real-time Interference-free Toolpath

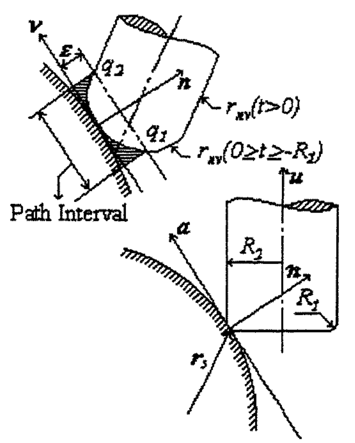


Fig. 4 : The determination of path interval

The key of CNC direct interpolation for surface machining is the generation of real-time interference-free toolpath. This algorithm must possess high real-time performance and stability. In various ways to generate real-time interference-free toolpath,[2-6] the way of surface model→cutter contact data → interference-free toolpath is comparatively suitable for generation of real-time interference-free toolpath, but its kernel is how to effectively check and handle the interferential CL point. On the basis of this point, a rather complete and high effective algorithm for generation of real-time interference-free toolpath has been designed,[7] which synthesizes many optimal measures. Its main procedure is shown in the Fig. 3.

3.5 Path Interval Processing

It is very important for the machining effiency and the surface quality to determine the suitable path interval. The factors which affect the path interval contain the cusp height, the geometric shape of the surface to be machined, the shape and dimension of the cutting tool, the feeding direction, etc. In this system, the enveloping theory is adopted to reasonably determine the path interval and the original surface near the current cutting point is replaced by the tangent plane at this point to simplify the calculation. In the realtime finish machining, the conclusion from the above algorithm is satifying because of the samll path interval.

For example, in the three-axis machining (as shown in Fig. 4), when the cutting tool moves towards the tangent line, the section contour of the enveloping body in the normal section $n-v$ is expressed as:

$$r_{nv}=\begin{cases}[u_v\cdot t-u_v\cdot u_n\cdot R/C+R_2\cdot(\cos\theta_1\cdot u_v\cdot u_n+\sin\theta_1\cdot u_a)/C]\cdot v+ \\ (R_1+R\cdot C+u_n\cdot t-R_2\cdot C\cdot\cos\theta_1)\cdot n \qquad t>0 \quad (\textit{thecylinderpart}) \\ \{u_v\cdot t-u_v\cdot u_n\cdot R/C+[R+(R_1^2-t^2)^{1/2}]\cdot[\cos\theta_2(t)\cdot u_v\cdot u_n+\sin\theta_2(t)\cdot u_a]/C\}\cdot v+ \\ \{R_1+R\cdot C+u_n\cdot t-[R+(R_1^2-t^2)^{1/2}]\cdot C\cdot\cos\theta_2\}\cdot n \qquad -R_1\le t\le 0 \quad (\textit{thetourspart})\end{cases} \tag{4}$$

where (u_a,u_v,u_n) is the coordinate component of the cutter axis u in the frame (a,v,n),

$R=R_2-R_1$, $C=(1-u_n^2)^{1/2}$, $\theta_1=arctg(u_a\cdot u_n/u_v)$,

$\theta_2(t)=\arcsin[u_a\cdot C\cdot t/(R_1^2-t^2)^{1/2}\cdot(u_v^2+u_a^2\cdot u_n^2)^{1/2}]-\arccos[u_v/(u_v^2+u_a^2\cdot u_n^2)^{1/2}]$.

In the curve r_{nv}, let $n=\varepsilon$ (the allowable cusp height), q_1 and q_2 can be calculated, then the path interval can be obtained from $|q_1q_2|$. For determining the path interval in terms of the whole-length of the toolpath, in this system, the 32 ms interrupting period is adopted to monitor the path interval for the smallest path interval increment.

3.6 The Feedrate Control

The aim of feedrate control is to ensure the dynamic precision of path and the quality of the surface to be machined as well as to control the feeding motion effectively. The control of feedrate contains two contents. They can be expressed as: (a) to determine the effective feedrate curve varying with the machining path. (b) to take some suitable measures of acceleration and deceleration for the smooth transition of feedrate.

3.6.1 The determination of the effective feedrate curve

(1) The effective feedrate of program segments should ensure the constant of the feeding to the part surface and the consistence of cutting conditions at any point of the machining path for the stable facial machining quality as far as possible.

The interference-free toolpath generated is the collection of discrete linear segmants. Both the discrete cutter location data and the ideal feedrate of this interpolating linear segmants are generated to ensure the constant of the feedrate . Let the programming feedrate be F, the feed of one program segment be Δr_{si}, the axial moving increments be $(\Delta r_{mi}(\Delta x_{mi},\Delta y_{mi},\Delta z_{mi}),\Delta\varphi_i,\Delta\theta_i)$, then the effective feedrate of this program segment can be given by: $F_i=F\cdot|\Delta r_{mi}|/|\Delta r_{si}|$.

(2) The limited condition of the effective feedrate

It is necessary for the dynamic precision of the toolpath to limit F_i according to the dynamic characteristics and the servo driving ability of the machine tool .

a. the limitation of the maximum speed. In a program segment, the axial effective speeds have to be smaller than the allowable maximum speed. For example, to the axis X, the limitation can be expressed as: $|F_{xi}|=|F\cdot\Delta x_{mi}/|\Delta r_{si}||<F_{x\max}$

b. the limitation of the maximum acceleration / deceleration .Because the direction and the value of synthetic feedrate of the two neighbouring linear segments may change, the variation in axial feedrates at their corner point should be in the range of servo driving ability to ensure the precision of the path. Let the feedrate of the endpoint in the segment i be F_{ie}, the feedrate of the starting point in the segment i+1 be $F_{(i+1)b}$. To the axis X, the limitation can be given by:

$$\begin{cases}|F_{xie}|\le|F_{xi}| \\ |F_{x(i+1)b}|\le|F_{x(i+1)}| \\ |F_{x(i+1)b}-F_{xie}|\le a_{x\max}\end{cases} \tag{5}$$

where $a_{x\max}$ is the allowable maximum accelerated speed/ decelerated speed.

c. the limitation of the distance for decelerating. A linear program segment has to be longer than the distance for decelerating in a decelerating rule when the feedrate of the starting point is higher than the feedrate of the endpoint in this segment. Otherwise, it is necessary to limit its feedrate of the starting point. The feedrate of corner point should be modified from end to beginning one by one to predict the limitation of the distance for decelerating in advance.

3.6.2 the smoothing of the feedrate curve

The feedrate of the starting point and the endpoint as well as the effective feedrate can be got after the above calculation. The index law which is more smooth is adopted to control the acceleration and the deceleration of the feedrate in this system (but the linear law is adopted to complete the acceleration and the deceleration of the auxilary approaching and retracting).

4. CONCLUSION

The programming is greatly simplified in the way of direct interpolation for surface machining. A high-level NC language similar to APT is adopted in the programming, not the collection of large discrete fine linear segments. In many cases, CAD data of surface can be directly applied to CNC machining. It will greatly cut down the auxiliary time and costs of surface machining for making surface finishing cheaper and more accuracy and efficient. Furthermore, the processing parameters including the dimension of cutting tool, the allowance for finish, the machining sequence, the structure and dimension of machine tool can be modified on line in this CNC system so that the operators can revise the machining process according to the actual situation at any time. All these are very important for raising the efficiency and quality of machining. The surface machining method based on the algorithm of surface direct interpolation has been realized in the Huazhong I CNC system. A batch of sample parts, such as integral centrifugal impeller, blade of air-blower, wave wheel of washing machine and other complicated compound surface, etc., have been successfully machined.

5. REFERENCES

1. Liu Huailan, Zhou Yanhong, Zhou Ji, “Machinery & Electronics”, Vol. 4, pp. 21-24, 1996 (in Chinese).
2. Bobrow J E, “NC Machine Tool Path Generation from CSG Part Representations”, Computer-Aided design, Vol. 17, No. 2, pp. 69-76, 1985.
3. Loney G C, Ozsoy T M, “NC Machining of Free Form Surfaces”, Computer-Aided Design, Vol. 19, No. 2, pp. 85-90, 1987.
4. Hwang J S, “Interference-free Tool Path Generation in the NC Machining of Parametric Compound Surfaces”, Computer-Aided Design, Vol. 24, No. 12, pp. 667-676, 1992.
5. Choi B K, Lee C S, Hwang J S, “Compound Surface Modelling and Machining”, Computer-Aided Design, Vol. 20, No. 2, pp. 127-136, 1988.
6. Choi B K, Jun C S, “Ball-End Cutter Interference Avoidance in NC Machining of Sculptured Surfaces”, Computer-Aided Design, Vol. 21, No. 6, pp. 371-378, 1989.
7. Liu Huailan, Zhou Yanhong, Zhou Ji, “Interference-free Tool Path Generation in the NC Machining of Parametric Surfaces”, China Mechanical Engineering, Vol. 7, No. 3, pp. 47-49, 1996(in Chinese).

SELECTION OF OPTIMAL MILLING PARAMETERS BY GENETIC ALGORITHM

T.L.Lau, Yang Qinghao
Department of Ind. & Mfg. Sys. Eng.
The University of Hong Kong
Hong Kong

ABSTRACT

This paper introduces an application of genetic algorithm in milling parameters selection and optimization. Mathematical model for milling parameters optimization has been setup. The processes of applying genetic algorithm in selection of optimal milling parameters are discussed in details. An example is shown to demonstrate the optimization process using genetic algorithm. The comparison between genetic algorithm solution and the analytical solution shows that genetic algorithm can be applied with standard procedures and can search out an approximate milling parameters close enough to the true optimum ones.

KEYWORDS

Milling, Cutting Conditions, Optimization, Genetic Algorithm

1. INTRODUCTION

Although the work on optimization of machining conditions had been proposed by CAPP working group of CIRP in 1966[1], much of the work has been done for turning[2,3], only few papers have been published for milling[4]. This is because the turning process can be simplified into a 2D problem. As milling process involves 3D geometry, different mathematical programming techniques have to be employed to handle a large number of process variables. However, mathematical programming methods have the following shortcomings:

1). Mathematical programming algorithm is case sensitive. Different equations or data formats may be used to describe the milling operation, such as tool life, cutting force, power available, tolerance and surface roughness. However, not every parameter can be expressed in equation format, but can be expressed as discrete data or in other formats such as neural networks, etc. The calculation procedures of mathematical programming must be modified according to different equations and data formats.

2). The calculation complexity of mathematical programming algorithm increases exponentially with the numbers of variable and constraint.

These deficiencies hinder the application of optimization in cutting conditions selection. At present, the cutting conditions are determined mostly based on process planner's expertise.

With the development of geometric modeling technology and computer technology, it is now the right time to consider both the geometric and technical information in order to select milling conditions systematically and optimally.

This paper explored the application of genetic algorithm in the selection of milling conditions. Through the comparison between the results of genetic algorithm and the analytical solution, it shows that the genetic algorithm can be applied with standard procedures and it can search out an approximate solution close enough to the true optimum value.

2. GENETIC ALGORITHM

Genetic algorithm is based on Darwinian survival-of-the-fittest evolution theories, only the fittest individuals in a population are likely to survive and generate offspring, thus transmitting their biological heredity to their new generations. The genetic algorithm emulates biological evolution theories to solve optimization problems. It comprises a set of individuals(a population) and a set of biologically inspired operators(selection, crossover and mutation). A genetic algorithm works with the following procedures [5]:

1). Initialize the first generation of population

2). Decode the individual's chromosome into a set of milling parameters.

3). Calculate the fitness value of each individual

4). Generate new generation by genetic operations.

a. Select two parents (the larger the fitness value, the higher the probability to be chosen as parent)

b. Crossover the chromosome of two parents at a probability P_c and generate two children.

c. For the newly generated children, mutate their chromosomes at a probability P_m.

d. Repeat steps a to c until the new population reach a predefined size.

5). Replace the old generation with the new generation, repeat steps 1) to 4) from one generation to another.

After thousand generations, most individuals in the population will be within a range of maximum fitness. The milling parameters corresponding to the chromosome of an individual, which has the maximum fitness value, will be an approximate optimal solution.

2.1 Code and Decode Optimal Variables

In simple genetic algorithm, every individual has an artificial chromosome represented in a binary string format, in which each fixed length of binary segment represents a variable in the application. The binary segment can be decoded into a variable's value. Mapping from coding a variable to a binary segment and vice versa is one-to-one correspondence.

Integer coding method uses a set of integers to code variables. Every integer has a value ranging from zero to a predefined maximum integer, which correspondent to a variable's range from its minimum to maximum. This method is widely used in engineering applications [6] and is also used in this paper.

2.2 Fitness Evaluation of Chromosome of individual

The fitness value of an individual determines its opportunity to be selected to reproduce its offspring. The bigger the fitness value is, the higher the chance the individual will be selected. This mechanism makes those with bigger fitness value inherit their chromosome to next generation. The reproduction process is actually an optimization process in genetic algorithm.

In engineering application, the optimal objective function can be evaluated by the decoded variables' value from an individual's chromosome. Minimizing or maximizing the objective(s) under certain constraint conditions is the purpose of optimization. If the objective is to maximize, the fitness value is equal to the objective value. If optimal objective is to minimize, the fitness value is equal to the inverse of objective value.

The optimal result must satisfy all constraint conditions. If an individual violates any one of the constraint conditions, the application instance of this individual represents is undesirable. In this case, a penalty value will be added to the fitness function to let the fitness value of this individual approaches zero. This makes those violating any of the constraint conditions have little opportunity to reproduce and to inherit their chromosomes.

2.3 Generation of New Population

The generation of new population includes three genetic operations: selection of parent, crossover and mutation.

2.3.1 Selection of Parent

Selection of parent ensure survival of those individual that has larger fitness value. Every time, two parents will be selected by roulette wheel slots method. To select a parent, all individual will line up in a closed circle. Every individual has a slot with angular value proportional to its fitness value.

2.3.2 Crossover operation

Up to a probability P_c, two selected parents will exchange part of their chromosome to reproduce their two children. Which part to be crossovered is determinated randomly. This means that the hybrid children could have better or worse fitness value. At a probability 1.0-P_c, two selected parents will copy their chromosomes to reproduce their children with identical chromosome. This means that the children will inherit the same chromosomes and fitness value as their parent.

2.3.3 Mutation operation.

For the newly generated child, every variable encoded in the chromosome has a probability P_m to mutate. If a variable is to be mutated, its encoded integer value will be selected randomly in its encoded integer range. This can reproduce brand-new individuals.

After selection of parents, crossover and mutation of chromosome, two children will be reproduced. These three genetic operations will be repeated until the new generation reach a predefined population size.

3. MATHEMATICAL MODEL OF MILLING CONDITIONS SELECTION

3.1 Variables

The milling parameters usually taken as variables for milling conditions optimization are as follows:

1). Cutting speed $V \in [V_{\min}, V_{\max}]$ in m/min;

2). Feed per tooth $S_Z \in [S_{Z\min}, S_{Z\max}]$ in mm/min;

3). Axial cutting depth $a \in [a_{\min}, a_{\max}]$ in mm.

The variables of radial cutting depth b and cutter parameters such as diameter of cutter D and number of teeth Z are also considered in some applications.

3.2 Constraint Conditions

1). The cutting force limit

The cutting force F_t should be less than a given value $F_{\max}$, which is due to torque available or the spindle motor capacity.

$$F_t = \frac{C_{F_t} \cdot b^{\beta_1} \cdot a \cdot Z \cdot S_Z^{\beta_2}}{D^{\beta_3}} \tag{1}$$

where,

$\beta_1, \beta_2, \beta_3$ are exponents and C_{F_t} is the cutting force coefficient.

2). The spindle power limit

The cutting power $P(n)$ at a spindle rotation speed n should be less than the maximum power $P_{\max}(n)$ that the spindle motor can provided at rotation speed n.

$$P(n) = \frac{D \cdot n \cdot F_t}{60 \cdot \eta} \le P_{\max}(n) \tag{2}$$

where,

η is power efficiency factor.

Other constraint conditions may be considered such as tool life etc. All constraint conditions can be expressed as a variable which is greater than or less than a limiting value or is within a given range.

3.3 Objectives

One objective of milling optimization is to minimize the cutting cost.

The cutting cost C for milling a length L of a workpiece is:

$$C = K_{c1}t_1 + K_{c2}(t_{21} + t_{22}) + K_{c3}t_3\frac{t_{21}}{T} + K_{c4}\frac{t_{21}}{T} \longrightarrow \min. \quad (3)$$

where,

$$t_{21} = \frac{L}{S} = \frac{L}{ZS_Z n} = \frac{\pi DL}{1000ZS_Z V}, \quad and \quad t_{22} = \frac{\pi Dl}{1000ZS_Z V} \quad (4)$$

$$T = \frac{C_T D^{\alpha_6}}{V^{\alpha_1} S_Z^{\alpha_2} b^{\alpha_3} a^{\alpha_4} Z^{\alpha_5}} \quad (5)$$

$K_{c1}, K_{c2}, K_{c3}, K_{c4}$ are cost coefficients. t_1 is the average nonproduction time per workpiece and $K_{c1}t_1$ is the nonproductive cost including loading and unloading etc.(This item can be ignored as it is not related to cutting parameter); t_{21} is the actual milling time that the cutter removes unwanted material and t_{22} is the cutter movement time such as cutter approach, departure and rapid traverse etc.; l is the equivalent approach and departure length; $K_{c2}(t_{21}+t_{22})$ is the milling cost; t_3 is the down time of machine tool due to changing of cutting edge and $K_{c3}t_3\frac{t_{21}}{T}$ is the tool change cost; $K_{c4}\frac{t_{21}}{T}$ is the tool(or inserts) cost that is evenly charged by the cutting time of the whole tool(or insert) life T; L is the actual milling length and S is the cutter feedrate in mm/min.; $\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5, \alpha_6$ are exponents in tool life equation and C_T is the tool life equation's coefficient.

4. EXAMPLE

The genetic algorithm is implemented for a face milling operation. The cutting geometry is a rectangular volume with 53mm × 16mm × 5mm (Length × Width × Depth). The workpiece material is CK45. The machine tool is Takisawa MAC-V2. The tool diameter is 25mm with 2 teeth. The inserts are SANDVIK R215.2-025 and insert change time is 5 min. The selection ranges of cutting parameters of this tool are: $V \in [50,250]$; $S_Z \in [0.05,0.40]$; $a \in [0.5,6.0]$. The cost coefficients are $K_{C2} = K_{C3} = 6.4$ HK\$/min. and $K_{C4} = 32.0$ HK\$ per insert.

For this application, the required axial cutting depth is within the permitted range of a, so the axial cutting depth is selected to be 5 mm and a single pass is needed. The optimizationparameters are {V, Sz}. The constraint conditions are maximum cutting force $F_{\max} = 13000N$ and maximum power available is calculated by:

$$P_{\max} = \begin{cases} 0.2238 + (5.595 - 0.2238)(n-100)/2500 & Kw, \ if \quad n \in [100,2500); \\ 5.595 & Kw, \ if \quad n \in [2500,7500]; \\ 5.595 - (5.595 - 3.73)(n-7500)/2500 & Kw, \ if \quad n \in (7500,10000]. \end{cases}$$

4.1 ANALYTICAL SOLUTION

To search the minimum cutting cost value, the partial derivatives $\partial C/\partial V, \partial C/\partial S_Z$ should be obtained from eq.(3)~(5) :

$$\left.\begin{aligned} \frac{\partial C}{\partial V} &= -\frac{C_1}{S_Z V^2} + (\alpha_1 - 1)\frac{C_2}{TS_Z V^2} \\ \frac{\partial C}{\partial S_Z} &= -\frac{C_1}{S_Z^2 V} + (\alpha_2 - 1)\frac{C_2}{TS_Z^2 V} \end{aligned}\right\} \quad (6)$$

where,

$$C_1 = K_{C2}\frac{\pi D(L+l)}{1000Z}, \quad and \quad C_2 = (K_{C3}t_3 + K_{C4})\frac{\pi DL}{1000ZT}$$

If $\partial C/\partial V = 0$; then $T = (\alpha_1 - 1)\frac{C_2}{C_1}$ and if $\partial C/\partial S_Z = 0$; then $T = (\alpha_2 - 1)\frac{C_2}{C_1}$.

As α_1 and α_2 is not equal in eq.(5), $\partial C/\partial V$ and $\partial C/\partial S_Z$ cannot be zero at same time, the cutting cost function of eq.(3) is a monotonic increasing function with V and S_Z. There will be no optimal cost value within the range of selected parameters under no constraint condition. The optimal value is on the boundaries of selection parameters or on the boundaries of constraint conditions.

For this application, the optimal solution is on the boundary of maximum power constraint condition. The analytical minimum cutting cost C_{min} and correspondent milling parameters are:

$V = 188.38$ m/min; $S_Z = 0.2069$ mm; $C_{min} = 0.5868$ HK\$.

4.2 GENETIC ALGORITHM SOLUTION

V & S_Z will be encoded into two integers range from 0 to $2^{16} - 1$.

$$I_V = \frac{V - V_{min}}{V_{max} - V_{min}} \times (2^{16} - 1), \quad and, \quad I_{S_Z} = \frac{S_Z - S_{Zmin}}{S_{Zmax} - S_{Zmin}} \times (2^{16} - 1) \quad (7)$$

The encoded integer set { I_V, I_{S_Z} } represents the chromosome.

The cost value C_f of each individual is expressed as:

$$C_f = C + M_P \cdot [\delta(F \le F_{max}) + \delta(P(n) \le P_{max}(n))] \quad (8)$$

where,

M_P is a positive value larger enough;

$\delta(x)$ is a logical function. If x is TRUE then $\delta(x)$=0, else $\delta(x)$=1.0;

If the spindle motor power and the cutting force are all under constraint conditions, then C is equal to C_f.

The fitness value of an individual is: $f = 1.0/C_f$ (9)

The population size in each generation is 50. The probability of crossover P_c is 0.5. The probability of mutation P_m is 2%.

Fig. 1 shows the average cutting cost of individuals in each generation. The graph of average cutting cost vs gemeration number indicates how fast the convergence is. It shows that after 30 generations, the average cutting cost can reach 90% of the analytical minimum cutting cost, and after 100 generations the average cutting cost reaches 95% of the analytical minimum cutting cost.

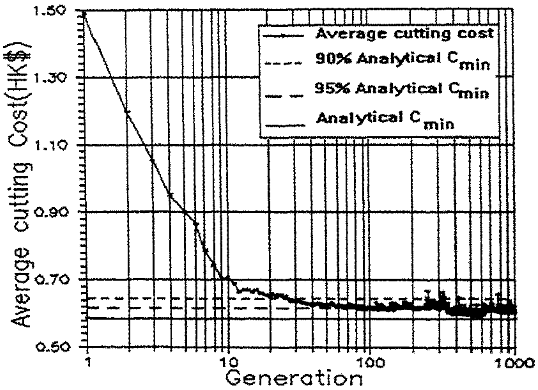


Fig.1 Average cutting cost of every generation

Table 1 shows twelve solutions of minimum cutting cost C'_{min} by running genetic algorithm twelve times. It shows that the error of average result $\overline{C'_{min}}$ is within 0.3% and the error of the maximum result of C'_{min} (the 8_th solution) is within 0.7% of the analytical minimum cutting cost C_{min} by genetic algorithm. For the selected milling parameters, the average error and the maximum error of selected value of S_Z are less than 0.5% of the analytical selected value of S_Z; the average error of selected value of V is within 1.8% and the maximum error of selected value of V is within 6.2% of the analytical selected value of V. As the cutting cost partial derivatives in eq.(6) shows that the value of $\partial C/\partial V$ is less than the value of $\partial C/\partial S_Z$ at the analytical optimization point $C_{min}(V, S_Z)$. The cutting cost value changes faster with the change of value of S_Z than with the change of value of V. Therefore, the error of selected value of V is larger than the error of selected value of S_Z.

Table 1. Twelve solutions of minimum cutting cost by genetic algorithm

	1	2	3	4	5	6	7	8	9	10	11	12
Sz	0.207	0.207	0.207	0.207	0.206	0.206	0.207	**0.206**	0.207	0.207	0.207	0.207
V	188.01	189 26	184.72	180.14	182.29	189.88	184.25	**176.70**	188.36	186.19	186 41	184.12
C_{min}	0.587	0.587	0.587	0.588	0.589	0.588	0.587	**0.591**	0.587	0.587	0.587	0 587
$\overline{C_{min}}$	0.588											

5. CONCLUSIONS

Selection of optimal milling parameters is one of the key procedures in detail process planning. An application of genetic algorithm in selection of milling parameters is presented in this paper. The example shows that the genetic algorithm can search out an approximate solution close enough to the true optimum value. In genetic algorithm, the optimization objectives and constraint conditions in machining operation need not be expressed in derivative format. The genetic algorithm can be applied with standard procedures and operates with integer values(except decoding variables and evaluating fitness value). This algorithm is more suitable to deal with different constraint formats which are usually the case in milling optimization.

Further research is needed to study the application of genetic algorithm in multipass milling optimization. Also, the effects of genetic algorithm's parameters(such as population size, crossover probability and mutation probability) on the convergence need to be studied.

6. ACKNOWLEDGMENTS

This research is sponsored by Croucher Foundation.

7. REFERENCES

1. Hoda A. ElMaraghy etc., "Evolution and Future Perspectives of CAPP", Annals of CIRP, Vol.:42 No:2,pp739-751, 1993.
2. Hinduja S. and Barrow G., "TECHTURN: a technologically oriented system for turned components", Proc. 1st Int. Conf. on CAPE, pp255-260, Edinburgh, 1986.
3. Ruy Mesquita, Evdovkia Krasteva, Stefan Doytchinov, "Machining Parameters in Multipass Turning", The Int. J. of Adv. Manuf. Tech., No:10, pp19-26,1995.
4. T. L. Lau, Optimization of Milling Conditions, Ph.D. Thesis of UMIST, Aug, 1987.
5. Daviad E. Goldberg, Genetic Algorithm in Search, Optimization and Machine Learning, p412, Addison-Wesley, 1989.
6. N. H. Wu, K. C. Chan, "A Genetic Algorithm Based Approach to Optimal Fixture Configuration", Computers & Industrial Engineering, Vol:31, No:3/4, pp919-924,1996.

CAM SOFTWARE FOR ULTRA-PRECISION ASPHERIC SURFACE USING TRI-ARC INTERPOLATION

M. Y. Yang and T. M. Lee
Department of Mechanical Engineering
Korea Advanced Institute of Science and Technology

ABSTRACT

This work is concerned with CAM software to enhance the precision and productivity of ultra-precision aspheric surface. The developed CAM software has input module which is adequate to ultra-precision aspheric machining and NC code generating module in which Tri-arc interpolation method makes tool path. In Tri-arc interpolation, the location of maximum interpolation error is fixed, so that tool path can meet tolerance fast and precisely. Generated NC code can be verified by dynamic simulation and analytical verification. The former makes it possible to inspect NC code visually and the latter checks invisible overcut by comparison of tool geometry with aspheric surface.

KEY WORDS

Aspheric Surface, CAM Software, Tri-arc interpolation, NC code Verification

1. INTRODUCTION

Nowadays, as consumer electronics, information, and aero-space industry grow, the demand for aspheric lens for camcoder, CD player, and optical instrument increases higher. The aspheric lenses have various shapes according to their application and often require several tens nanometer in the form accuracy. Thus, to enhance the productivity and precision of aspheric surface, CAM software for ultra-precision aspheric surface needs to be realized.

In this study, CAM software which is suitable to the ultra-precision machining and minimizes the error of tool path is proposed. The proposed CAM software can improve the productivity and precision of aspheric surface.

This CAM software has input modules for various aspheric surfaces, diamond tool geometry, cutting condition, and so on. Especially in the input module for cutting condition, it is necessary to give the several contour cutting condition to avoid bad surface roughness due to machining affected layer.

In ultra-precision machining, the relative motion of tool and workpiece is precisely copied onto the workpiece[1)]. Accordingly, interpolation of aspheric surface must meet the allowable tolerance exactly. In fitting aspheric curve for CNC machining, linear interpolation generates huge amount of NC codes for satisfying the very small tolerance, and makes scallops due to the acceleration and deceleration of tool in every linear motion. On the other hand, arc interpolation generates a shorter NC code than linear interpolation does. It can make the 1st differential value to be continuous at the connection points of each arc[2)], so that it can eliminate scallops. However, the existing arc interpolation needs a large amount of calculations to meet tolerance exactly. For the fast and precise calculation of error between the arc and the aspheric curve, we developed Tri-arc interpolation method in which the location of maximum error is fixed.

In this CAM software, NC code verification module is required to prevent an overcut and a collision between tool and machine. These troubles can be found by dynamic simulation and analytical verification. Dynamic simulation make it possible to verify NC code visually. Analytical verification can check invisible overcut by comparison of the radius of curvature with tool nose radius.

2. INPUT OF CAM

To obtain NC code for ultra-precision aspheric machining, one must input the figure of aspheric surface, the geometry of diamond tool, the information on raw stock, and cutting condition for rough and contour cuttings.

Aspherical surfaces can be classified into axisymmetric and non-axiaymmetric. Our research limited the figure of aspheric surface to axisymmetric which general ultra-precision NC machine can produce[3)]. In our CAM software, plane, spherical surface, and aspheric surface described by parametric equation can be specified, and for user's convenience, parabola, ellipse, and hyperbola can be specified(See Table. 1).

Table.1 : Equation of machinable figure in the CAM

SURFACE	EQUATION	PARAMETER
Plane	•	•
Sphere	$Z^2 + X^2 = R^2$	R
Aspheric Surface by Aspheric Parameter	$Z = \frac{cX^2}{1+\sqrt{1-(1+K)c^2X^2}} + AX^4 + BX^6 + CX^8 + DX^{10}$	c,K,A, B,C,D
Parabola	$Z = a \times X^2$	a
Ellipse	$\frac{Z^2}{A^2} + \frac{X^2}{B^2} = 1$	A, B
Hyperbola	$\frac{Z^2}{A^2} - \frac{X^2}{B^2} = 1$	A, B

The geometry of R-type diamond tool which is exclusively used for ultra-precision aspheric machining can be specified in our CAM. The geometry such as tool nose radius, tool angle, rake angle, and relief angle is utilized for NC code generation and interference check between tool and workpiece.

In terms of machining process, raw stock for aspheric surface can be classified into cylindrical type and offset type in which rough cutting was already executed. In case of cylindrical type, NC code for all cutting processes from rough cutting to finish cutting should be obtained. In case of offset type, NC code except rough cutting should be obtained. So, these requirements are satisfied in our CAM.

In ultra-precision machining, to avoid bad surface roughness due to machining affected layer, several contour cuttings are indispensable. So, in the input module of cutting condition, it is possible to input cutting condition for several contour cuttings.

3. TRI-ARC INTERPOLATION

To generate tool path precisely and fast, a new circular interpolation method called a Tri-arc, in which a segment span is composed of three circular arc and the maximum error of interpolation is fixed, is used.

In the Tri-arc method, the first arc(S_1) is generated from the start knot point(P_1) to the position (P_3) where the maximum error occurs. If a junction P_3 exists on the normal vector at the point P_v which is the farthest point from the chord $\overline{P_1P_2}$ and its tangential direction runs parallel with the chord $\overline{P_1P_2}$, the maximum error exists at the junction P_3(See Fig.1).

To support the above hypothesis, the error between the original curve and generated arc at arbitrary position (P_a) is analyzed as given below.

$$|error| = r_1 - \sqrt{x^2 + f^2(x)} \tag{1}$$

$$|error|' = -\frac{x + f(x)f'(x)}{\sqrt{x^2 + f^2(x)}} \tag{2}$$

Thus, the extremum satisfies the equation

$$x+f(x)f'(x)=0 \tag{3}$$

The above equation can be rewritten as

$$\frac{f(x)}{x}f'(x)=-1 \tag{4}$$

In equation (4), $\frac{f(x)}{x}$ means the slope of $\overline{O_1P_a}$. The extremum exists at a position where the tangent vector at a point on the original curve is perpendicular to the radial direction from the corresponding point on the generated arc, in other words, where the normal vector at a point on the original curve passes through a center of the generated arc. Therefore, junction P_3 has the maximum value of the interpolation error and knot P_1 has the minimum value of that in Fig.1. These results are the same to the right side that will be generated.

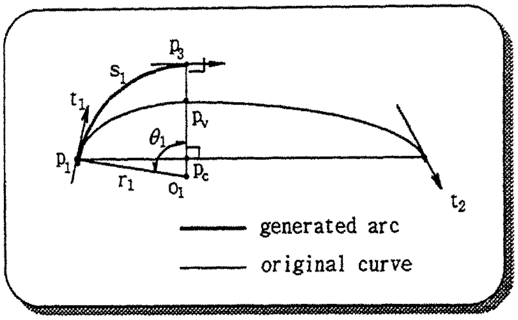


Fig.1 : Generation of the First arc in Tri-arc interpolation.

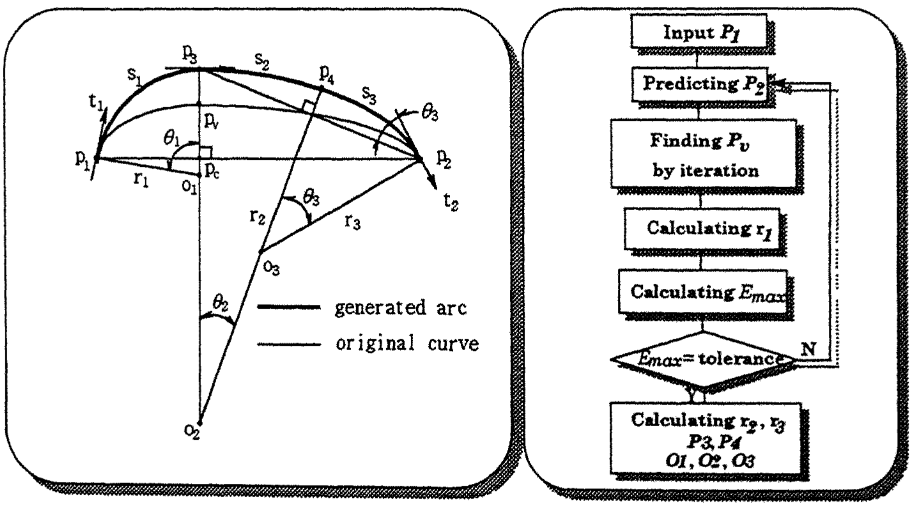


Fig.2 : Geometric configurations and flow chart for Tri-arc interpolation.

To generate the second and third arc, six conditions are required.

- - The second arc(S_2) must pass through the point P_3.
  - The secend arc(S_2) must have tangent vector at the point P_3 parallel with the chord P_1P_2.
  - The third arc(S_3) must pass through the point P_2.
  - The third arc(S_3) must have tangent vector parallel with the chord P_1P_2 at the point P_2.
  - The two arcs must have a point of common tangency.
  - The normal vector at the point P_4 is perpendicular to a straight-line P_3P_2 which connects the first junction P_3 and the terminal knot P_2.

According to the above process, Tri-arc method generates three arcs between two knots as shown in Fig.2. The advantage of Tri-arc method is that the maximum error can be calculated without error iteration in contrast with general arc interpolation method. Therefore, it is easy to make the error close to the allowable tolerance by controlling terminal knot P_2(See Fig.2). The error maps for various aspheric profiles show that Tri-arc method meets given tolerance exactly(See Fig.3).

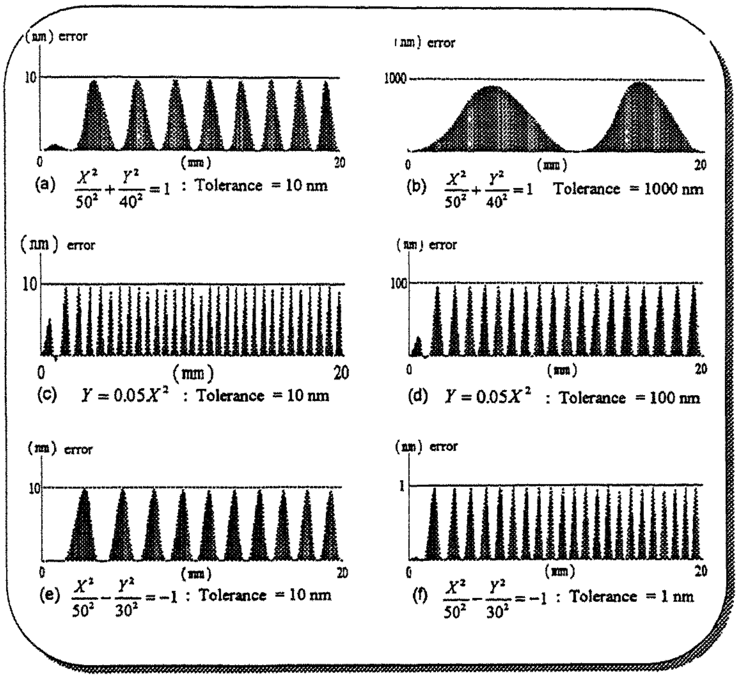


Fig.3 : Error map between Tri-arc and various aspheric profiles.

4. NC CODE VERIFICATION

To verify tool path, our CAM software can perform dynamic simulation and analytical verification. The former makes it possible to inspect NC code visually and the latter checks invisible overcut by comparison of tool geometry with aspheric surface.

Dynamic simulation performs imaginary machining with input data in the screen. Although user must examine tool path with the naked eye, it is easy to judge whether interference occurs or not and whether depth of cut and feedrate is proper or not(See Fig.4). However, visual examination has limit to find fine interference, so that it should be used along with analytical verification.

In analytical verification, following fine interferences can be inspected, that is, smaller radius of curvature of aspheric surface than the tool nose radius(R)(See Fig.5(a)), smaller tangent angle(θ_w)of aspheric surface than the tool angle(θ_t) (See Fig.5(b).), and smaller tangent angle(θ_w) of aspheric surface than another tool angle(θ_r)(See Fig.5(c)).

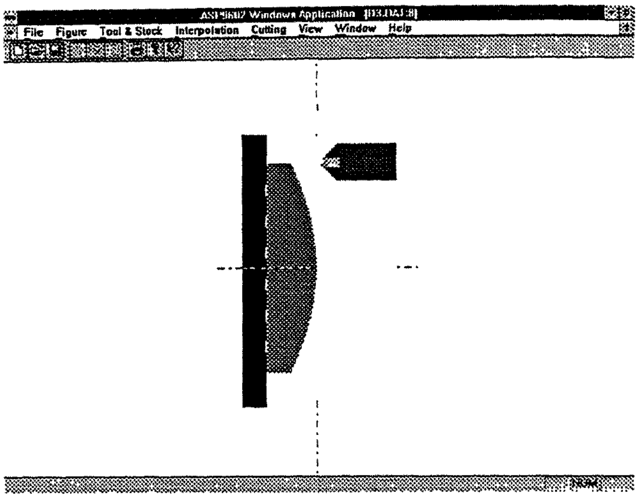


Fig.4 : Dynamic simulation for the verification of tool path.

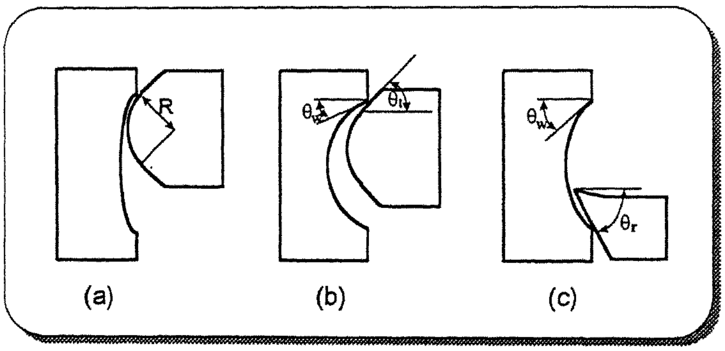


Fig.5 : Inspectable interference between tool and workpiece.

5. STUCTURE OF CAM SOFTWARE

The entire structure of CAM software consists of input part, process part, and output part. In input part, aspheric surface, tool, stock, and cutting condition are specified. Process part calculates tool path, generates NC code, and verifies NC code. Finally in output part, input data and NC code can be modified, saved, and transmitted to the machine.

To realize the entire structure, CAM software has the following menu structure(See Fig.6). In the 'Figure' menu, aspheric surface is specified. In the 'Tool & Stock' menu, information on the geometry of tool and stock is given respectively. The method of interpolation is selected in the 'Interpolation' menu, and cutting condition is specified in the 'Cutting' menu. After finishing all processes for input, NC codes are generated by clicking the 'NC Code Generation' menu. Through the 'Report' menu subordinated to the 'View' menu, all input data are displayed, and through the 'NC code' menu, NC codes are displayed. In the 'Simulation' menu, user can verify NC codes. Finally, NC code can be transmitted to machine through the 'Communication' menu(See Fig.6)

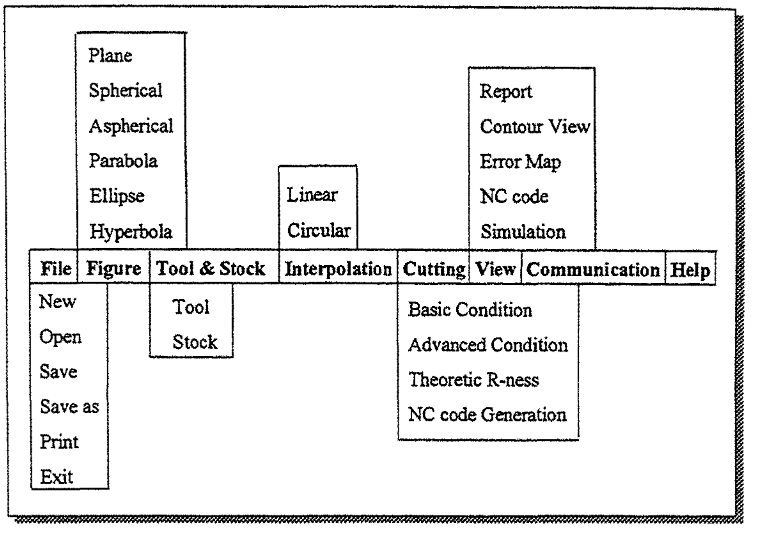


Fig.6 : The entire menu structure.

6. CONCLUSION

Summarizing briefly, we developed CAM software for ultra-precision aspheric surface which has the following characteristics.

- Aspheric surface by aspheric parameter which includes parabola, ellipse, and hyperbola can be input. In the input module for cutting condition, several contour cuttings can be specified to avoid bad surface roughness.
- To generate tool path precisely and fast, Tri-arc interpolation method, in which the location of maximum error is fixed, was developed. Tri-arc method generates tool path about five times faster than a general Bi-arc method.
- Through dynamic simulation which performs imaginary machining in the screen and analytical verification which checks invisible overcut, interference can be inspected.

The proposed CAM software can be expected to improve the productivity and precision of aspheic surface.

7. REFERENCE

1. N. Ikawa, Ultraprecision Metal Cutting - the Past, the Present and the Future, pp. 589-594, CIRP Vol. 40, 1991.

2. D. B. Parkinson and D. N. Moreton, Optimal biarc-curve fitting, pp.411-419, Computer-aided design, Vol. 23, No. 6, 1991.

3. M. A. Gan and N.I.Kulikovskaya, Aspherical surface in optical instruments, pp.653-660, The Optical Society of America, 1991.

Cutter Location Data (CLdata) Generation for Flat Tool in Multi-axis Milling Operations

Khalil Aslam Awan
B E(Mech); MSc(UK), DIC(UK); PhD(UK)
Mechanical Engineering Department
The Papua New Guinea
University of Technology
Private Mail Bag
Lae, Papua New Guinea
Tel. (+675) 473 4853 Fax. (+675) 475 7017

ABSTRACT

The advanced techniques using computers like Finite Element Method (FEM), Boundary Element Method (BEM) etc. of designing a mechanical component made the shapes of these components very complex. It is possible for the design engineer to use these state of the art tools to produce designs with highly complex free form surfaces. The typical shapes include turbine blades, volute, ducts, dies, aircraft and automobile bodies etc. To manufacture these highly complex and some times doubly curved surfaces is a real challenge for manufacturing engineer. The Computer Numerical Control (CNC) machine technology is advanced enough to manufacture these surfaces, but to generate the manufacturing information and its verification requires highly complex computer algorithms. Most often these complex surfaces are not possible to manufacture by using 3-axis milling machines. The manufacturing engineers need the freedom of 5-axis machining for producing these profiles.

The main problem with 5-axis machining is the generation of collision free machine tool paths and their simulation for checking collision and gauging or under and over cuts. The flat ended cutting tool is ideal for getting the best surface finish as the cutter edge of the tool is fully exposed to the machining surface. The face of the tool closely follows the surface so the cusp hights are minimum. The size of the data generated for the tool motion is also minimum as less number of passes are required.

In this research paper a computer algorithm is presented which generates the collision and gauge free flat tool cutter path for a 5-axis milling machine. The comparative analyses of the flat end tool and radius end tool is also given to show that in the flat end tool less manufacturing data is generated and less manufacturing time is required.

The algorithm is implemented on an IBM compatible PC computer by using C programming language. the system can interactively generate the manufacturing information for the flat end tool. It is possible to alter the tool path according to the users wishes. The output from the system is in the form of collision free cutter location data (CLdata) which contains the tool position and its orientation. CLdata may be passed through a post processes filter for a particular CNC machine to generate the machine specific cutting tool motion instruction.

KEYWORDS

Flat ended tool, Multi-axis milling, CAD/CAM, Tool offset.

1. INTRODUCTION

The use of Numerical Control (NC) machine tools for the manufacturing of mechanical components is now common practice. To manufacture sculptured surface components by using a milling cutter with the help of these types of machines can impose many problems, like from where should the cutting start, how deep should be the depth of cut, what path should be followed to cut the shape accurately etc.

Conventionally a manufacturing engineer or part programmer writes a programme manually for the NC machine tool with the help of an engineering drawing which s/he (she or he) may have obtained from the design engineer. After writing and loading a programme s/he verifies it either by cutting a test part or by using a plotter to plot the tool paths. If s/he finds that his written programme is satisfactory then s/he can produce the component. It is quite evident that this whole process can take months depending on the complexity of a particular job [Besant86].

This whole process of "manual" involvement of a part programmer in the writing and producing mechanical components is very time consuming and the chances of error are very high. Therefore, this method is being rapidly replaced by the representation of mechanical parts by some mathematical or geometrical model. Moreover, the design and manufacturing activities are seem as integrated process aided by new computing techniques. This means CAD can readily interface with CAM resulting in NC machine tool programs, be it on tape, disk, or preferably down loaded to a CNC machine using networking. The main advantage of this approach is that the design or manufacturing engineer can generate all the information and verify it on the graphical terminal of his work station.

With the use of this technology, the production engineer can produce the verified and accurate tapes of a complex mechanical component without leaving computer terminal. This whole process can increase the speed and reliability of the design to manufacture life cycle which means saving in cost and time saving in information transfer.

In the NC machining of sculptured surfaces over cutting, also called gouging, is one of the most critical problem. In NC machines the tool is moved by giving the coordinates of some reference point of the tool (usually center of the ball end or flat end mill tool), this point is called Cutter Location data (CLdata) point. The real contact point between the tool cutting surface and the sculptured surface to be cut is called Cutter Contact (CCdata) point. CC data point is usually different than the reference CLdata point. Due to this arrangement it is very easy to over cut the surface as shown in Fig. 1.

A Satisfactory handling of cutter interference requires not only the avoidance of over cutting, which inevitably results in an under cutting, but also subsequent removal of the undercut volume by using smaller diameter ball-end cutters (see Fig. 1)

In NC machining of sculptured surfaces starting from a surface model, there are three commonly used approaches for generating CL (Cutter Location) data.

1. Surface Model to Cutter Contact (CC) data to Cutter Location (CL) data.
2. Surface model to Polyhedral model to CL data.
3. Surface model to offset-surface model to CL data

In the first method as far as cutter interference is concerned, Method may either produce over cut-free CC data or generate CC data without taking into consideration the tool interference and then converting this data into interference free CL data [Bobrow85] [Gould72] [Loney87]

[Kim88a][Kim88b]. The possible drawbacks with this method might be that it involves extensive numerical iteration which are not always stable and it is difficult to remove selectively the undercut volumes using smaller diameter ball-end cutters.

The second method is very robust in that it prevents the cutter from gouging the polyhedron without numerical stability problems [Duncan83][Duncan77][Choi89]. It also provides the means for the subsequent removal of the undercut volumes using smaller flat-end mill cutter.

The offset surface approach [Choi88] [Kral86] [Faux79] has some distinctive advantages as it allows more flexibility in cutter pass planning If the surface to be machined is a rectangular parametric C^2 surface (i.e. surface with positional as well as slop continuity), then its offset surface becomes C^1 (i.e. surface with positional continuity only) so that a surface-surface intersection algorithm [Barnhill87] can be employed in identifying interference regions. An excellent treatment of the parametric offset surface is provided in [Forouki86]. This approach deserves further investigation

2. NEW APPROACH

A surface patch can be defined by using two parametric directions u and v, both u and v vary from 0 to 1 over the surface as shown in the Fig. 2. This surface patch can be divided into a grid of m x n nodes by giving equally spaced intervals of u and v all along the surface. In this way a surface can be approximated by a grid of m x n nodes. If the spacing between the u and v intervals are small then the points obtained on the grid represents the actual surface within a reasonable tolerance.

The surface can be divided into small triangular planes by joining the diagonals of the grid m x n either in the direction shown by solid lines in Fig. 2 or by the direction shown in the doted line i.e. the surface is approximated by a number of these triangular planes within a tolerance. The whole surface can be considered as triangular faces or facets of a polyhedron.

For the manufacturing of this surface, a tool should be placed on each point of the nodes m x n in turns. The sequence of this placement of the tool on the nodes will depends on the direction of machining.

Duncan [Duncan83] used the similar method but in his method he places the tool in such a way that it touches the triangular planes at the centroid of the triangle to be machined and avoids the other triangular planes by calculating the distances between the tool surface and the other neighboring triangular planes.

The main disadvantage of the Duncan's method is that if the tool is placed at the centroid of each triangular plane then the tool path would not be smooth, rather it will make a zigzag motion as shown in Fig. 3. On the other hand in our proposed method, the tool is placed on the node points of the polyhedron and tool always follows these points. To check the gouging with the neighboring triangles it will check for collision with them. If there is collision it will be avoided by displacing the tool along the tool axis direction at the nodal point, hence the tool will always remain at the top of the nodal point.

In the milling operation three types of cutting tool can be used as shown in Fig. 4 abc [Gaal87].

1. Ball end cutter.
2. Flat end cutter.
3. Radius cutter.

The machining operation can be done in two modes.

1. Face milling.
2. Side milling.

In face milling of sculptured surfaces, the tool will be along the normals of the surface where as in the side milling the tool is in the tangential direction to the surface. In this work we will concentrate on the offset calculation of flat end cutter.

2.1 Face milling with a flat end cutter

In this research paper, method for tool offset calculation for flat end cutter is given. The procedure of finding the offset for ball end cutter is given in [Awan95] and radius end cutter will be given in a separate research paper. The reference point of the cutter i.e. the center of the end has to be off set as shown in Fig. 4b.[Gaal87]

From Fig. 4b.

$$\mathbf{T} = \mathbf{C} + R(\mathbf{n}_t \times \mathbf{C}_n \times \mathbf{n}_t)$$

where

$\mathbf{T}$ = Tool reference point.
$\mathbf{C}$ = Cutter location point on the surface.
R = Radius of cutter.
$\mathbf{C}_n$ = Unit normal vector on the surface at C.
$\mathbf{n}_t$ = Unit normal vector along the axis of the tool.

The normal vector and the point **C** on the surface is known from the surface definition on each node. $\mathbf{n}_t$ is known from the orientation of the tool axis.

After displacing the tool to point **T** it may still be possible that the tool will interfere with some surrounding triangular plane facets. Consider Fig. 5 in which the surface directly under the tool (i.e. the surface which will be under the tool if we look along the tool axis directly) is triangulated. The triangular planes, which are directly under the tool, will only effect the position of the tool, hence gouging with these planes can be avoided by moving the tool in its axis direction.

Consider a typical point **C** where the tool has to be placed, the surrounding triangular facets which may effect the position are shown in the Fig. 5.

Consider an imaginary plane p_C having a normal along the tool axis direction i.e. $\mathbf{n}_t$ and passes through point **C**. The neighboring triangular facets may interfere with the tool only when at least one of the vertices of a particular triangular facet is above the plane p_C.

To find out whether a point $\mathbf{P}_A$ of center of the triangular facet is above a plane p_C or not consider Fig. 6.

Equation of the plane p_C is given by

$$\mathbf{r} \times \mathbf{n}_t = \mathbf{C} \times \mathbf{n}_t = \mathbf{p}$$

where

$\mathbf{p}$ = Perpendicular distance of plane p_C from origin.
$\mathbf{r}$ = Any vector from origin to plane p_C.

Calculate $\mathbf{P}_A \times \mathbf{n}_t - \mathbf{p}$

If

$\mathbf{P}_A \times \mathbf{n}_t - \mathbf{p} > 0$ Then the point is above i.e. in the positive direction side of the plane normal $\mathbf{n}_t$.

$\mathbf{P}_A \times \mathbf{n}_t - \mathbf{p} = 0$ Then the point is on the plane.

$\mathbf{P}_A \times \mathbf{n}_t - \mathbf{p} < 0$ Then the point is below i.e. in the negative direction side of the plane normal $\mathbf{n}_t$.

So if any of the vertex of the triangles around tool is above the point **C**, that triangular facet might interfere with the tool Otherwise if all the vertices of a triangular facet are below the point **C**, the facet will be interference free.

Consider any arbitrary triangular facet ABC as shown in the Fig. 7. We want to find out whether this triangular facet will interfere with the tool or not when it is placed at point **C**. One or more vertices of this facet are above the point **C**. Consider point **T** (offset point) and draw a normal from **T** to plane ABC which will cut plane ABC at J as shown in the Fig. 8.

The unit normal vector on the plane ABC is given by

$$\mathbf{N} = \frac{(\mathbf{B}-\mathbf{A})\times(\mathbf{C}-\mathbf{A})}{|(\mathbf{B}-\mathbf{A})\times(\mathbf{C}-\mathbf{A})|}$$

The distance $|\mathbf{TJ}|$ is given by

$$|\mathbf{TJ}| = l - |\mathbf{T}.\mathbf{N}|$$

where

l = Perpendicular distance between plane ABC and origin.

l = $\mathbf{A}.\mathbf{N} = \mathbf{B}.\mathbf{N} = \mathbf{C}.\mathbf{N}$

If $|\mathbf{TJ}| > \mathrm{R}$ (R = tool radius) tool will not interfere with facet ABC.

If $|\mathbf{TJ}| < \mathrm{R}$ then the tool might interfere with the plane ABC, but it may not be within the triangular bounds of the facet fixed by the coordinates of the vertices **A**,**B** and **C**.

Now

$$\mathbf{J} = \mathbf{T} + |\mathbf{TJ}|\mathbf{N}$$

and from Fig. 8.

$$\mathrm{v} = \sqrt{\mathrm{R}^2 - |\mathbf{TJ}|^2}$$

If v < R or less than the normal distance between **J** and line **AB**, **BC** or **CA** than the tool will interfere with the facet even though point **J** is outside the triangle ABC. Also if J is inside ABC than interference will occur.

By using the above procedure [Duncan77] it is possible to determine whether a facet will interfere with a tool or not.

Consider Fig. 9 in which a two dimensional view along the plane ABC is shown. To avoid the tool interference, the tool has to be moved in the direction $\mathbf{n}_t$ until the perpendicular distance between the plane and the tool reference point **T** is equal to or greater than R. This new position of the tool is **T'** as shown in the Fig. 9.

Consider a unit normal vector along **TK** direction

$$\mathbf{e} = (\mathbf{n}_t \times \mathbf{N}) \times \mathbf{n}_t$$

So

$$|\mathbf{TK}| = \frac{|\mathbf{TJ}|}{\mathbf{e} \times \mathbf{N}}$$

If R > |TJ| and K is within triangle ABC then the tool will interfere with the plane facet.

where

$$\mathbf{K} = \mathbf{T} + \mathbf{e}|\mathbf{TK}|$$

from Fig 9. we have

$$\tan\theta = \frac{\mathrm{R} - \mathbf{TK}}{\mathrm{d}}$$

$$\mathrm{d} = \frac{\mathrm{R} - \mathbf{TK}}{\tan\theta}$$

So the new interference free position of T is

$$\mathbf{T}' = \mathbf{T} + \mathrm{d}\mathbf{n}_t$$

This procedure should be repeated for every facet under the projected circle of the tool and T' corresponding to the maximum value of d should be taken as the interference free value

3. TURBINE BLADE EXAMPLE

The above offset generation algorithm was used to manufacture an axial flow turbine blades as shown in for 10. The triangular facets are not shown for simplicity of graphics. Note the offset generated by the above algorithm on the surface for ball end cutter. In this particular case the axis of cutting tool ($\mathbf{n}_t$) is required to be in the tangential direction to the surface for the good cutting condition during machining. In fig 10 only one pass of the tool is shown on the blade, after this cut the tool will move down into the blade to generate a similar path After careful study of the graphics and results obtained by computer it was evident that the tool is not over or under cutting the surface of the blade.

This new approach was developed with in a brooder research program. The aim of the program was to design and implement a computer software (DMCS-5D) system for the designing and manufacturing of complex surfaces. The method has some advantages than the other methods like [Duncan83][Duncan77][Choi89]. The calculations are less computationally intensive and the tool path generated by this method is far more smoother than the other approaches. It is also possible to implement this approach in any of three, four or five axis machining cases as the orientation of the tool will not effect the calculation of tool offset by this method. The approach is using triangular facets as the surface approximation, the tool interference is calculated for all the effecting triangular facets so the problem of gouging or over cut can very effectively be avoided by the use of this method.

4. REFERENCES

[Barbhill87] R. E. Barbhill ; M. Jordon ; G Farin ; B. R. Piper, *Surface/surface intersection.* Computer aided geometric design. Vol 4. 1987 pp 3-16.

[Besant86] C. B. Besant. and C. W. K. Lui, *Computer aided design and manufacture.* 3rd Ed. Ellis Horwood.

[Bobrow85] J. E. Bobrow, *NC machine tool path generation from CSG part representations.* Computer aided design Vol 17. No 2. 1985 pp 69-76.

[Choi88] B. K. Choi ; C. S. Lee ; J. S. Hwang ; C. S. Jun, *Compound surface modeling and machining.* Computer aided design Vol 20. No 3. 1988,pp 127-136.

[Choi89] B. K. Choi and C. S. Jun, *Ball-end cutter interference avoidance in NC machining of sculpture surface.* Computer aided design Vol 21.No 6,July/Aug 89.

[Duncan77] J. P. Duncan and S. G. Mair, *The anti-interference feature of polyhedral machining.* Advances in Comp. aided manuf. D McPherson ed. North Hollent Pub. Co.

[Duncan83] J. P. Duncan and S. G. Mair, *Sculpture surface in Engineering and Medicine.* Cambridge University Press.

[Farouki86] R. T. Farouki, *The approximation of degenerative offset surfaces.* Computer aided geometric design Vol 3. 1986 pp 15-43.

[Faux80] I. D. Faux; M. J. Pratt, *Computational Geometry for design and manufacture.* Ellis Horwood Ltd. 1980.

[Gaal87] B Gaal, *Automatic Generation of NC data for 5-axis milling from a solid modeler.* ACME Cambridge Univ. 1987.

[Gould72] S. S. Gould, *Surface programs for numerical control in curved surface engineering.* Churchill College, Cambridge.

[Kim88a] Kwangsoo Kim and John. E. Biegel, *An Integrated approach to sculpture surface design and manufacture.* Comp. Ind. Engng Vol 14. No 3. pp 271-280.

[Kim88b] Kwangsoo Kim and John. E. Biegel, *A path generation method for sculpture surface manufacture.* Comp. Ind. Engng Vol 14. No 2. pp 95-101.

[Kral86] Irvin. H. Kral, *Numerical Control Programming in APT.* Prentice Hall.

[Loney87] G. C. Loney and T. M. Ozsoy, *NC machining of free form surfaces.* Computer aided design Vol 19. No 2. 1987 pp 87-90.

[Awan95] wan K. A. *Tool Offset Generation for ball end cutter in Multi-axis Machining Operations.* The International Conference on Industrial Engineering and Production Management, Marrakech, Morocco, April 4-7, 1995.

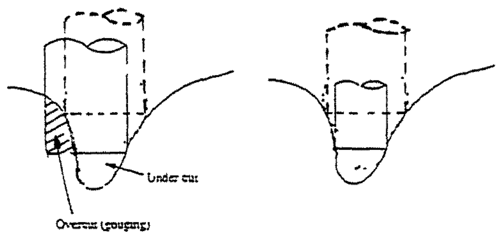


Fig. 1. Cutter interference (gouging) in NC machining.

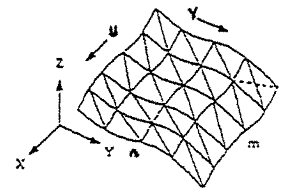


Fig. 2. Triangulation of surface to a polyhedron.

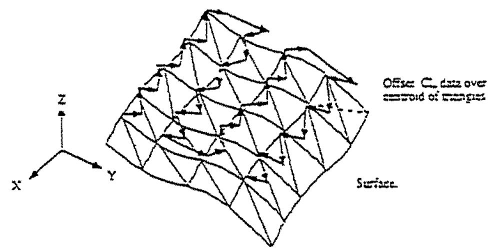


Fig. 3. Offset tool motion over triangulated façets.

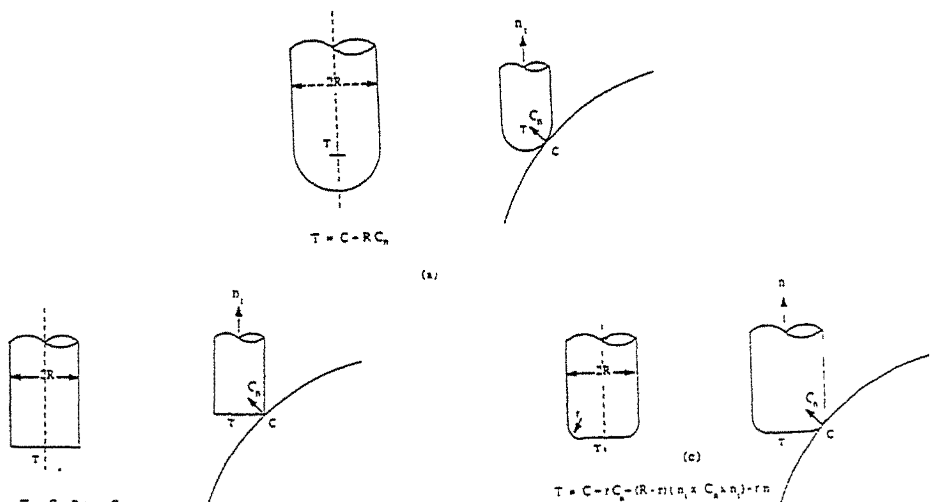


Fig. 4 Cutter Contact (CC) point vs Cutter Location (CL) point for (a) Ball (b) flat and (c) end radius milling tools

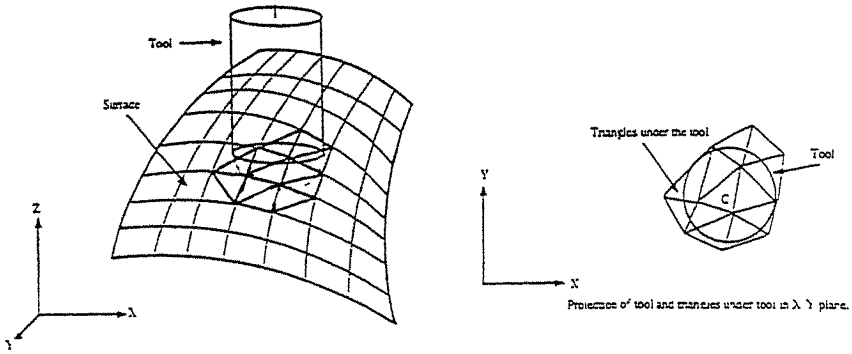
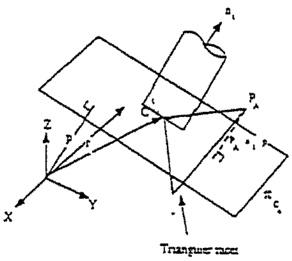


Fig. 5 Tool and triangles under the tool that will effect the tool interference calculation



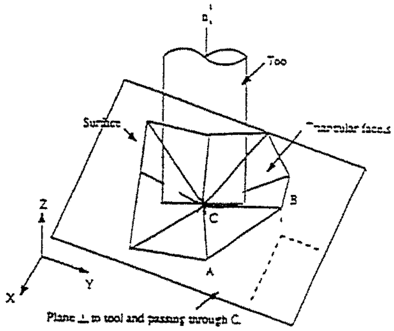


Fig. 6. Position of a point P_a relative to a plane π_c. Fig. 7. Triangular facets and tool interference

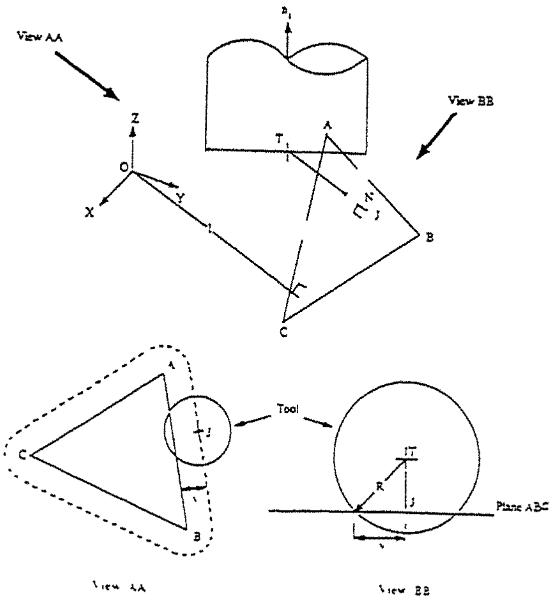


Fig 8 Triangular facet's distance from the tool

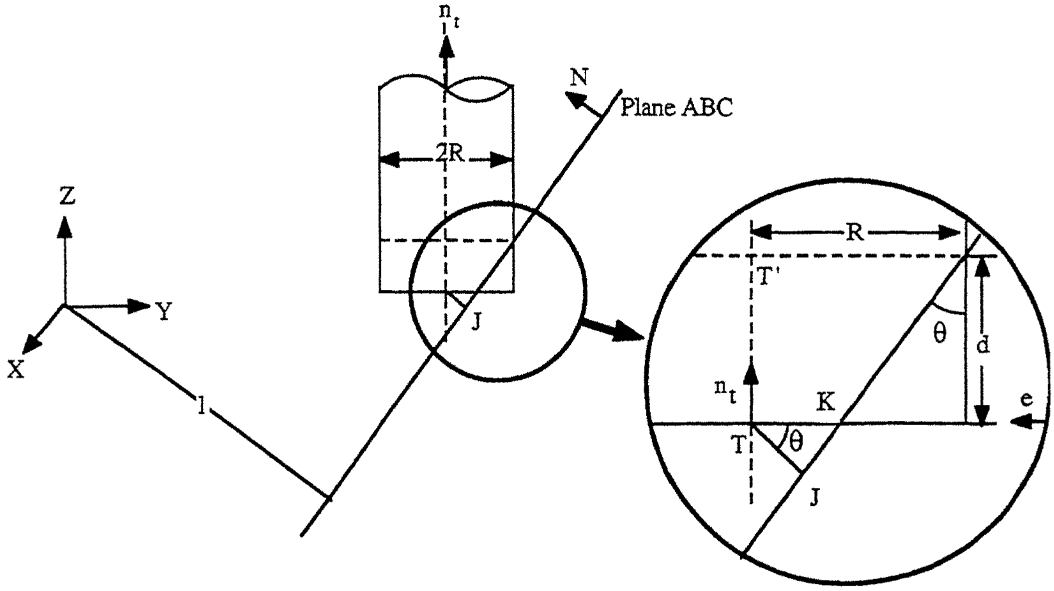


Fig. 9. Interference avoidance of flat end cutter.

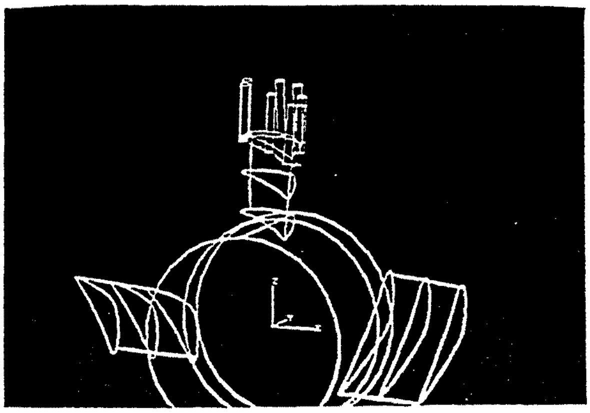


Fig. 10. Machining of the Blade.

POCKET MACHINING NC PROGRAM VERIFICATION BASED ON PLANE PARTITIONING

O. Ben Ahmed Daho, J.F. Poiraudeau and M. Bazzi
LICN, Université de LIMOGES
Allée André Maurois, 87065 LIMOGES Cedex, FRANCE
E-mail:poiraudeau@unilim.fr

ABSTRACT

Computer modelling and simulation technology assists in validating NC (numerical control) programs before they are executed on an NC machine tool. This technique is known as NC verification; it consists in detecting automatically the errors, without recourse to human judgement This paper describes a method applied to NC programs, which provides a complete representation of all machined regions of the part. The method also determines the number of times (history of region) these regions are machined. Each region is an overlapping of some areas swept by the cutting tool. The aim of our study is to determine the histories of all the regions; indeed, repeated machining increases machining time, and may also decrease the surface quality. The algorithm is implemented, and a practical example of a pocket with islands obtained by CAD-CAM system is presented.

KEYWORDS

NC Verification, Plane Partitioning, Computational Geometry, History of Region.

1. INTRODUCTION

As is the case with computer programs, NC programs may contain errors of several types, ranging from the relatively trivial syntax errors to subtle interactions with work-holding fixtures. However, when the machining environment is known, NC verification provides the means for detecting certain errors in NC programs before they are executed on the shop floor, thus significantly shortering the product development cycle [1]. Most of the research on NC verification has been focused on the nominal spatial effects of machining. A new approach was recently proposed [2]. It consists in the automatic verification of NC programs where the automatic error detection replaces the user visual inspection.

This paper presents a method which can be applied to NC programs in order to check if the following requirement is respected: regions of the part should not be machined repeatedly, as repeated machining might decrease the surface quality and increase machining time. In the case of pocket machining (also called 2 1/2D milling), the algorithm presented here makes it possible to determine the complete representation of all the machined regions as well as the number of times these regions are being machined. This study, which is a part of our research made on virtual machine tools, is an extension of a work presented in [3]. In order to tackle efficiently the great number of toolpaths which may be present in an NC program, we have used a computational geometry method: plane partitioning [4]. It can be implemented by using the data structure called planar map [5]. The problem addressed here is building a data structure to support the incremental insertion of a new toolpath in planar map, dynamically computing new intersections, and updating the data structure. Some research into the pocket machining problem has already been done, relying on the concept of computer geometry called Voronoi diagrams for the toolpath generation [6].

We build a partitioning for each pocket with a given depth. Each partitioning cell describes a machined region of the pocket. In order to determine how many times a region is machined, each region is taken as an overlap shared with a set of swept areas. Furthermore, each cell is associated with a counter called history, which indicates the number of swept areas. To our knowledge, no research has yet been done into the quantification of the overlap of areas. We present an experimental study based on the extreme example (worst case analysis) and the realistic example of pocket machining with several islands ; the realistic example is obtained by using a CAD-CAM system.

2. POCKET MACHINING

The majority of industrial machining tasks consists of pocket machining: the removal of material within a defined boundary contour. Due to the fact that a large number of mechanical parts are *terrace-shaped* or can be approximated to that, a 2 1/2D type of control is often used. The pockets are realized by several machining passes with different cutter tools of cylindrical form ; the rotation direction is here not considered. The pockets can contain several positive islands. We use a file as an input which is generated by **LICN** [1] software where each block or cycle contained in the NC program is developed in a set of elementary toolpaths (linear or circular). The swept area is the area enclosed by the envelope curve generated by the cutter as it travels from the starting point to the end-point of the given toolpath (see Figure 1). This envelope curve is then modelized by a single four-edged polygon. An edge can either be a straight line segment or a circular arc. Thus, the pocket profiles and the islands contours are built from the straight line segments and the circular arcs.

The NC program is incrementally interpreted. For each new depth, a partitioning is built for the pocket which will be machined. The latter is induced from a set of polygons associated with the toolpaths which are needed for machining this pocket. The polygons are defined as closed parametric curves, this induces an order notion between points of the same polygon. The parametrization is realized by the increasing numbering of polygon points beginning from zero. Two main types of milling strategies are employed in CAD-CAM systems: direction-parallel milling and contour-parallel milling.

3. BUILDING OF PARTITIONING

Much research into plane partitioning by curves has been carried out, especially work by H. Edelsbrunner and B. Chazelle, who proposed incremental methods for inserting curves such as straight lines and circles [4]. However, their research is based on restrictive assumptions as the intersections of lines or circles, if non empty, consist of two points at most. In the industrial application presented here, the functional toolpaths are consecutive. Therefore, the swept areas which are then obtained overlap, and the associated polygons will have overlapping edges. Figure 1 illustrates the case of a zigzag pocket machining.

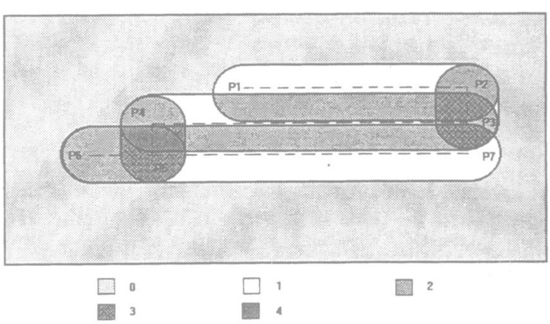


Fig. 1 : Swept areas.

[1]The LICN software, emulator of NC controllers, is a registered trademark of Limoges University.

The building of the partitioning $\mathcal{P}(n+1)$ consists in determining all the possible intersections between the new polygon P_{n+1} to be inserted and the polygons P_j $(0 \leq j \leq n)$ which are part of the partitioning $\mathcal{P}(n)$. Each polygon inserted in a current partitioning is stored in a list which is associated with the partitioning and sorted chronologically. In this section, we consider that each polygon to be inserted intersects with at least one inserted polygon.

3.1 Data Structure

Plane partitioning is represented by a planar map whose elements are of the vertex, arc and cell types. The well-known winged-edge structure is used to modelize the planar map. The representation includes on the one hand, topological information (adjacent relations between the elements of the map), and on the other hand, geometrical information (coordinates of the intersection points associated with the vertices). The association of point and vertex, piece of polygon and arc, polygonal contour and cell is then immediate. Overlapping pieces of polygon share the same arc. Modifying the topology of partitioning is realized by applying two Euler operators (insertion of a vertex in an arc, and cut of a cell according to an arc). The planar map representation is thus maintained during all the steps of the building the partitioning.

3.2 Polygon Incremental Insertion

The initialization of the partitioning consists in inserting the first polygon. An arc bordering two cells with the same polygonal contour must be created. One of the cells represents the machined area, the other -the silhouette cell of the partitioning- represents the rest of the material. At this stage, the partitioning will be described by a planar map with a single arc and two cells.

Inserting a new polygon is achieved by following intersections. We determine a first intersection between the polygon to be inserted and one of the polygons P_j $(0 \leq j \leq n)$. The intersection points that are found are ordered according to the parametrization of the polygon P_j and are stored in a list associated with it. Let the arc $a(P_j, u_0, u_1)$ be cut by P_{n+1} at a point A_{t_0} with the parameters t_0 for P_{n+1} and u for P_j. When $u_0 < u < u_1$, a vertex s_{t_0} corresponding to A_{t_0} can be created and inserted, thus dividing the arc a into two arcs: $a(P_j, u_0, u)$ and $a(P_j, u, u_1)$. When $u = u_0$ or $u = u_1$, s_{t_0} can be initialized with the extreme vertex of the arc a corresponding to u. Then, thanks to the localization procedure, the first cell c cut by the polygon P_{n+1} is found. The possible intersections between the polygon P_{n+1} and the contour of the cell c are determined. The intersection point A_{t_1} which is closest to A_{t_0} according to the parametrization direction on the polygon P_{n+1}, is stored. Let $a(P_k, v_0, v_1)$ be the arc of the cell c cut by the polygon P_{n+1}. The vertex s_{t_1} corresponding to the point A_{t_1} is inserted and the arc $a(P_k, v_0, v_1)$ is divided into two arcs (except when $v = v_0$ or $v = v_1$). The arc $a(P_{n+1}, t_0, t_1)$ is then created and inserted binding the two vertices s_{t_0} and s_{t_1}, and the cell c will be cut in two according to the arc. However, when the two polygons P_{n+1} and P_j have pieces of polygons which overlap between the two vertices s_{t_0} and s_{t_1}, the cell c will not be cut. This process is continued by following intersections until back to the first intersection point initially found.

3.3 Localizing cells

Localizing cells consists in determining a cell among the cells adjacent to the one which has been cut by the polygon to be inserted. Such a cell is determined by exploiting the planar map structure once the intersection point and the piece of polygon on which it is situated are known. The cell on the left (or respectively on the right) of the arc corresponding to the piece of polygon is selected if the cell on the right (or respectively on the left) of the arc is cut. Localizing the first cell cut by a polygon to be inserted consists in determining whether the polygon to be inserted cuts the cell on the left or on the right of the intersecting arc ; this can be done by travelling on the polygon from the first intersection point. We thus obtain a pair of intersecting arcs.

4. REPEATED MACHINING AND THE CONCEPT OF HISTORY

The planar map of a partitioning associated with a given pocket provides a complete representation of the machined areas and of the possible positive islands of the pocket. To determine how many times a region is machined, each region is taken as an overlap shared with a set of areas swept by the different tools. In order to quantify the areas for each region, we introduce the idea of cell history. Each cell is associated with a counter called history, which indicates the number of swept areas that overlap. A cell of history 0 therefore describes a non machined area of the pocket (positive island). A cell of history three for instance corresponds to three passes of the cutting tool in the same region associated with the cell. Commonly, when trajectories are generated in the CAD-CAM system, the distance between one tool-path and the next is chosen so as to obtain an overlap. Such an overlap will be associated with the regions of history two.

When the partitioning is initialized, the corresponding planar map is composed of two cells. One is associated with the first machined area which is given the history one. The other, the silhouette cell which describes the rest of the material, will always keep the history 0. When inserting a polygon P, the new cells thus created are given the same histories as the cut cells. The new cells situated on the left of the pieces of the polygon P by moving on it counter clockwise are stored in a temporary list C. The cells which are not cut by the presence of overlapping edges are also stored in this list. The polygon may also include other cells without cutting them. To determine the cells, we use the adjacent relations between the cells of the list C. Updating the histories will consist in incrementing the histories of the cells in the list. The incrementation represents the pass of the cutting tool once more in the regions associated with the cells. In othet words, we give a semantic interpretation of the building process of the partitioning where a history is attributed to each cell.

5. MAP NON CONNECTEDNESS

So far we have considered that each new polygon to be inserted must intersect with at least one polygon belonging to the current partitioning. This condition enables us to achieve the finding of the first intersection and to obtain a planar map with a single connected component. However, the machining of a pocket may involve several cutting-tool retractions (in the case of a pocket with monotonous areas). This problem could be solved by building a new partitioning at the same depth as the current partitioning, for every insertion failure. A partitioning would thus be associated with each monotonous area of the pocket. However, the monotonous areas of the same pocket always present overlaps. As a result, the histories of the regions where the areas overlap are erroneous. One way of overcoming this difficulty is to create a list associated with the current partitioning, in which we store polygons that have failed to be inserted. After each successful insertion, the polygons of this list are reintroduced in the building process. At the end of the processing of the machining program, if the subsidiary list of a partitioning at a given depth is non empty, the possible partitionings entailed by the polygons from this list are built again.

6. EXPERIMENTAL RESULTS

The implementation of the algorithm has been realized on a 9O MHz Pentium personal computer in the C language. A graphical visualization shows how the machining evolves at each new toolpath and how the history of the region changes. In order to evaluate its practical performance, we selected two examples. The first two are obtained with a commonly used CAD-CAM system and realize the same pocket with four positives islands. The direction-parallel milling strategy is used in the first example, and the contour-parallel milling strategy in the second one. The ratio between the pass distance and the diameter of the tool is 0.6. In the first example, the machining is performed with five monotonous areas. These areas are the sets of consecutive functional toolpaths separated by the tool retraction.

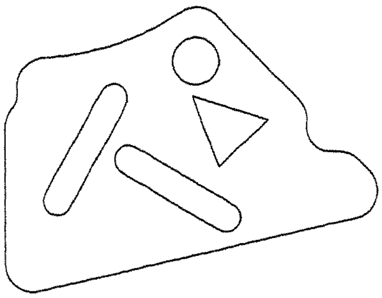


Fig. 2.a : Pocket profile.

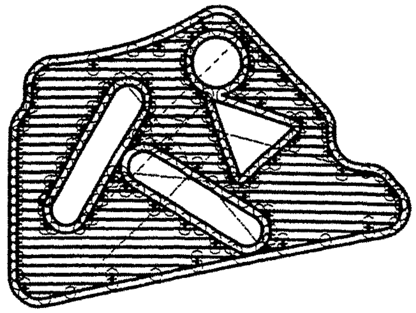


Fig. 2.b : Direction-parallel milling.

Table 1 presents the number of vertices, arcs and cells as well as the CPU time of the algorithm for the first example.

Tool path	Vertices	Arcs	Cells	CPU time (s)
143	701	1299	600	1.76
254	1240	2303	1065	5.78
258	1266	2353	1089	5.91
300	1475	2743	1270	7.52
335	1656	3079	1425	8.36

Tab. 1

For the first two examples, we computed the measures of the areas and of the pocket. In table 2 we give the distributions of the measures of the areas which have the same history compared with the measure of the pocket surface. Thanks to this results, it would be possible to estimate the surface quality of the pocket after the machining.

Direction-parallel milling	**History**	0	1	2	3	4	5	6	> 6
	Cell number	4	252	455	389	275	47	2	0
	Percentage	16.48	22.41	47.03	7.08	6.60	0.32	0.04	0.0
Contour-parallel milling	**History**	0	1	2	3	4	5	6	> 6
	Cell number	4	261	384	306	245	28	6	0
	Percentage	16.48	27.02	31.61	13.51	9.23	1.05	0.7	0.0

Tab. 2

In the third example which is unrealistic, the program performs a 25 linear toolpaths set like the prongs of a rake, we then cut them with toolpaths perpendicular to the prongs. This example holds a high number of intersections for a limited number of toolpaths. In this case, the performing time is shorter than in the first example even though they both involve approximately the same number of elements in their respective planar map. This is due to the fact that in the realistic example the search for the first intersection of the polygon to be inserted is performed on a connected list which can contain a high number of polygons.

25 tool paths cut by	Vertices	Arcs	Cells	Islands	CPU time (s)
1 tool path	150	300	152	24	0.18
5 tool paths	550	1100	552	120	0.32
10 tool paths	1050	2100	1052	240	0.55
15 tool paths	1550	3100	1552	360	0.78
20 tool paths	2050	4100	2052	480	1.03
25 tool paths	2500	5100	2552	600	1.29

Tab. 3

7. CONCLUSION

In this paper, we have tried to determine the machined areas of a pocket and the number of times those areas are being machined. Hence, we have adopted the method of plane partitioning so as to obtain a complete representation of the areas. We then introduced the concept of history associated with each cell of the pocket partitioning. The history of a cell gives the number of times the area of the cell is machined by the cutting tool. Thanks to this method, it would also be possible to optimize the cutting parameters (feedrate and speed) by using toolpath segments. Indeed, the available data structure (while taking into account the direction of rotation), allows to compute subsegments where the radial width remains constant with a change of the tool feedrate. By relying on computational geometry concepts, the method presented here can be extended from the verification of NC programs to their optimization.

References

[1] Bernard Chazelle & al "Application Challenges to Computational Geometry CG Impact Task Force Report". Technical Report TR-521-96, Princeton University, April 1996

[2] J P. Menon and H B. Voelcker "Toward a comprehensive formulation of NC verification as a mathematical and computational problem" *Journal of Design and Manufacturing*, 3:263–277, 1993.

[3] J. F. Poiraudeau and O. Ben Ahmed Daho. "Reconstruction de profils de poches pour la validation des programmes d'usinage en commande numérique". In *Deuxième Conférence Internationale sur l'Automatisation Industrielle*, volume 1, Nancy, 7-9 Juin 1995.

[4] F.-R. Degott. *"Partitionnement du plan par des courbes"*. PhD thesis, Université Joseph Fourier, Grenoble, Décembre 1991.

[5] M. Gangnet, J.-C. Hervé, T. Pudet, and J.-M. Van Thong. "Incremental Computation of Planar Maps". Technical report, DEC, Paris Research Laboratory, Mai 1989.

[6] M Held, G. Lukacs, and L. Andor "Pocket Machining Based on Contour-Parallel Tool Paths Generated by Means of Proximity Maps". *Computer-Aided Design*, 26(3):189–203, March 1994.

INFLUENCE OF TOOL PREPARATION SCHEDULING ONTO THE PERFORMANCE OF MACHINING CENTRES

G. Petuelli and U. Mueller
Department of Mechanical Engineering
University of Paderborn at Soest
Germany

ABSTRACT

This paper describes a simulation based system which was developed in order to improve the availability and utilisation rate of metal cutting machines, especially of flexible manufacturing systems (FMS). As will be explained, the Event-Oriented Tool Management System (ETM) introduced enables planning and scheduling of tool preparation, e.g., tool and cutter grinding and resharpening. The scheduling is based on production plans, the procedure incorporated will be explained. Any activity to be performed in order to make tools ready for use as well as resources available, e.g., manpower and tools and tool and cutter grinding machines, is taken into account. The performance of the simulation module is assessed based upon the data of a FMS installed in industry, such as workpieces to be machined and tools to be provided in order to guarantee maximum output.

KEYWORDS

Tool Servicing, Simulation, Scheduling, FMS

1. INTRODUCTION

The performance of machining centres and flexible manufacturing systems to a great deal depends upon the availability and the quality of the tools needed to machine parts according to the production plan. Different evaluations showed that tooling to a large amount influences the productivity of manufacturing units [1]. Reduced productivity is caused by, e.g., missing tools, delayed assembly of tools and arrangement of tool kits, tool breakage and limited life of cutting edges as well as the supply with faulty tools. Extensive research work was done and it was found that up to 30 % of machine failures are caused by missing tools and the supply with faulty tools [2]. Thus, the machine availability is reduced by about 4 %. This figure does not include down time caused by the exchange of tools due to tool wear or tool breakage. Because of these influences the machine utilisation rate is reduced by at least another 6 %, in total leading to a loss of 10 % production capacity. Most importantly, the evaluation also indicated, that the influences were caused mainly by poor organisation of tool flow and tool servicing.

Nevertheless, most of the efforts made to improve the performance of manufacturing systems are concentrated onto the development tools to optimise the layout and design of manufacturing systems and facilities, as well as production planning and scheduling procedures. To a certain degree planning and scheduling software tools are available [3] and have been applied successfully [4]. Major research work is carried out to improve the performance of software needed to set-up simulation models of manufacturing systems [5,6]. On the other hand, efforts have been made to support scheduling and sequencing of workpieces [7] as well as scheduling of machining operations by aid of process planning systems [8]. No provision is made in any system to take into account tool servicing, which means, resharpening as well as tool preparation, e.g., exchange of indexable inserts, and the time involved to make tools ready for use again, once their life is exhausted.

A tool planning and scheduling system introduced was developed for a particular application and can be understood as a stand-alone unit [9]. This system does not include any information about grinding technology to be applied in tool preparation, and does not feedback information about work in process, and the status of particular cutting tools to the planning department to assure the schedule of machining of workpieces. The tool selection problem with respect to optimisation of the number of tools is described and evaluated theoretically by aid of neural networks [10].

On the other hand, the performance of tool data management systems has to be improved, which provide information about tool geometry and to some extend about their usage, respectively technological data allocated for particular applications [11]. Information covering tool servicing must be integrated, since, up to now, the systems do not include any information or activity related to re-sharpening and preparation of tools.

2. INTEGRATION OF PRODUCTION PLANNING AND TOOL SCHEDULING

In order to overcome the limitations discussed above the Event-Oriented-Tool Management (ETM) was developed [12]. The basic idea of this software tool is to integrate data exchange between the different departments of a company involved in manufacturing, taking into account the special requirements of tool servicing, as indicated in figure 1. The system enables interaction of and data exchange between production planning modules and planning and scheduling procedures provided by ETM. For this purpose an existing tool data base is upgraded in a way such as to insert any data covering tools, planning of their usage, tool flow and reconditioning as well as resharpening. This includes tool and cutter grinding machines, grinding wheels, grinding technology and any device used in tool rooms and tool shops, e.g., tool presetting units.

As already mentioned, ETM is simulation based and it basically consists of two major components, respectively simulation modules. One module enables the description and the set-up of simulation models of manufacturing facilities such as flexible manufacturing systems (FMS), whereas the second component is built to enable simulation and scheduling of tool servicing, tool and cutter grinding and resharpening. This simulation system is based on the procedures and control features provided by SLAM II. Special modules have been written to integrate material flow and information flow of both components.

Information generated and available in production respectively manufacturing and tool shop or tool servicing area are linked via ETM. This link enables the evaluation of tool lists to define the type of tools and the point and time they are supposed to be available and ready for use at the machines. It is based on the sequence of workpieces to be machined and work orders. Distinction is made between tools to be resharpened and tools which have to be prepared, e.g., at which inserts have to be exchanged.

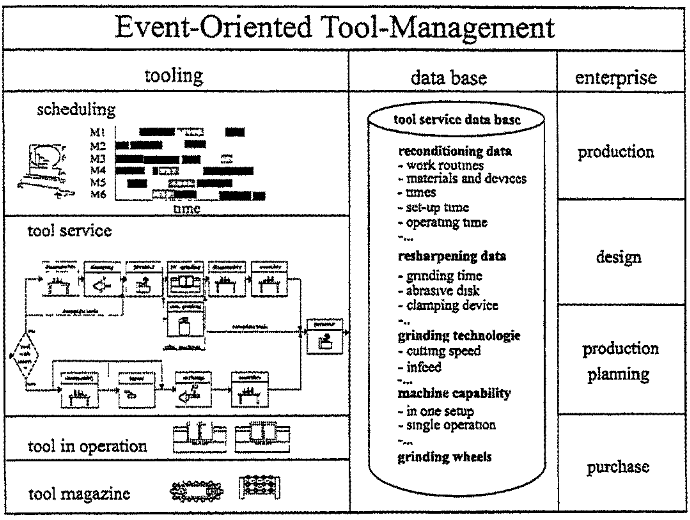


Fig. 1: Extended data base to integrate information flow by ETM

The interaction of the two above mentioned components of manufacturing facilities enables the optimisation of manufacturing systems in total. This covers not only, e.g., the selection of machining centres, material handling systems and scheduling of workpiece machining to determine the performance of manufacturing systems, but also tool flow and tool scheduling for servicing purposes. Bottlenecks in manufacturing as well as in tool servicing are determined by the consideration of the inter-dependencies between workpieces machining and tooling of machines, which are depended on the performance of tool shops.

3. DATA BASE AND SCHEDULING PROCEDURE

The interaction of different components provided for tool servicing purpose and described in appropriate simulation models is defined by rules, which are stored in a special data base [13]. In figure 2, the structure of the data base of tool and cutter grinding machines is depicted to give an example of the basic conception of the different elements or modules. As can be seen, machine descriptions basically consist of five major parts, the most important one is labelled "processing data".

This section of the data base includes information about machine capabilities and their performance. References are given to tools that can be handled by the machine, as can be seen on the right hand side of the figure. Additional elements of the data base are related to grinding technology, tool geometry and material, tool servicing rules, including constraints and resources such as grinding wheels and tool clamping units.

Initial conditions of a scheduling process are given by the sequence of machining of workpieces and the corresponding tool lists in conjunction with due dates of tools. The scheduling procedure takes into account constraints given by machines and devices and other resources, their availability and performance, as well as by processing data and grinding rules.

As already mentioned, processing data are most important with respect to tool preparation, since they establish the relationship between machines, grinding technology and tools. The combination of processing data with the definitions and rules according to which the individual tool might be serviced, are basic information to simulate and schedule tool service.

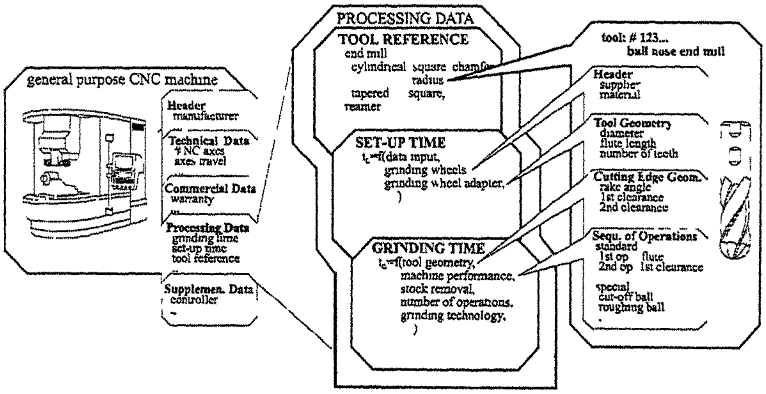


Fig. 2: Interaction of tool data base and grinding machine data base

The procedure to prepare the individual tool and its flow through a particular tool room is determined in three phases. In the first step the operations necessary to service the individual tool either are defined by visual examination of the tool or by aid of preliminary data and statistics about the tool wear expected. The appropriate rules can be retrieved from the tool data base by aid of the identity number of the tool. Depending on this information the tool flow in general is defined. The tool will be directed to the different departments of the tool room.

Furthermore, the kind of wear defines whether a standard or a special service must be applied. Standard service means that a given sequence of grinding operation must be followed in order to resharpen a tool according to the requirements of the end-user. The sequence is stored in a particular section of the tool data base. Special service is related to, e.g., excessive wear or even excavation of the cutting edges. Some of this kind of case-based rules are included in the data base and can be modified by the user of the system, additional rules may be defined.

In the second phase the alternative processing sequences evaluated are defined by the allocation of the operations to machines and devices available and capable to perform the individual operation. For this purpose, cross references between different components of the data base are defined (fig. 2). The operation or machining times of the alternatives are calculated in the third phase prior to simulation of planning and scheduling. The calculation of resharpening times is based upon technological data that can be realised by the machines or devices available, the performance of grinding wheels, the wear of the cutting edges and the tool geometry as well as the surface quality required. The time needed to prepare a tool with indexable inserts is determined based upon the operations to be performed and the number of inserts to be exchanged.

The schedule is defined by the comparison of the results of different simulation runs, which are done in order to evaluate the performance and productivity of alternative servicing sequences using different machines and devices. Since tooling and tool handling are quite dynamic processes it is of utmost importance to update planning and scheduling of tool servicing according to the needs of manufacturing. New simulation runs can be initiated by different events, one of which is the transfer of worn-out tools to the tool servicing and preparation area. On the other hand, the time frame covered by a planning and scheduling run, respectively the number of tools to be scheduled, was defined in a way such as to cover part of the work load of manufacturing only, thus, a new scheduling run can be initiated by a predefined number of tools out of this time frame, the remaining stock. By these definitions the execution time of simulation and scheduling could be reduced by about 90 % [14].

Finally, the schedule that covers a particular period of the work orders, is transferred to the production and manufacturing planning and control department. Whenever tardiness of a particular tool is forecast by the tool room it is up to the planning department to initiate appropriate actions to meet the production plan, e.g., to order spare tools, define a new workpiece processing plan using different tools, or to rearrange and reschedule the manufacturing plan.

4. APPLICATION

As already mentioned, poor tooling to a large degree influences the performance of manufacturing machines. Consequently, it is important to provide and keep in stock the optimum number of tools necessary to assure productivity and rate of utilisation at the highest level possible.

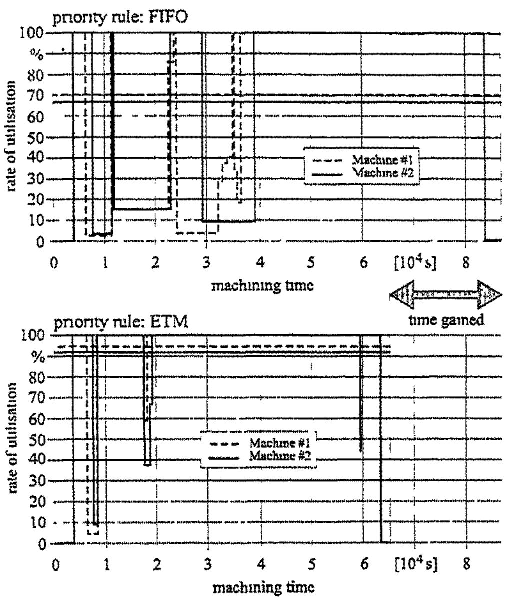


Fig. 3: Influence of priority rule onto the utilisation rate of the FMS

The standard procedure to define the number of tools necessary is based on trial and error, it is minimised since capital investment is to be minimised [2]. Regarding, e.g., the installation of FMS, generally, the number is defined by two samples of each type of tool needed to perform the different operations in the first instance. This procedure is quite simple and does not lead to a qualified start-up of FMS with respect to their performance. On the other hand, tooling of existing manufacturing systems is based upon the operations to be performed, the capacity of tool magazines of machining centres, and the experience of the person who is in charge of planning of tooling. This in most cases leads to an excessive number of tools in stock, since in most companies there are no means available to monitor and track status and whereabouts of tools. The Event-Oriented-Tool Management enables the optimisation of the number of tools and guarantees maximum output of the production as will be shown by aid of data of a FMS.

The FMS under consideration consists of two machining centres, a workpiece handling system, workpiece loading/unloading stations, buffer or temporary storage, and a measuring machine. Furthermore, the machining centres are linked to and by a tool handling system. The layout as well as the sequence of the workpieces to be machined was optimised by aid of simulation. Tool servicing is done in a tool preparation area next to the tool room.

Based on the sequence of the workpieces and the work orders, the type of tools and the point and time they are used are evaluated, lists are generated which define tool kits. Finally, the simulation run of tool servicing is started to calculate the number of tools needed to machine the workpieces. Preconditions were given in a way such as to allow the purchase of an unlimited number of tools whenever there is no tool available to perform a machining operation. By this assumption the performance and rate of utilisation of the FMS, as well as the minimum number of each kind of tool to be provided is evaluated. The initial conditions for the sequential simulation runs are improved by carefully examining the utilisation rates of the machining centres and the throughput of the tool preparation area. The optimum number of tools to be provided was determined to 105 tools of different kind, whereas by the application of the standard procedure there would have been 166 tools to be purchased.

Based on the optimum number of tools the effects caused by the sequence in which the tools are prepared was examined. In the first instance scheduling was done by using the priority rule *first-in-first-out* (FIFO), whereas in a following scheduling procedure the tools were serviced according to the requirements of the FMS, as evaluated by ETM. The results of these different simulation and optimisation runs are depicted in figure 3. The first diagram (top, priority rule FIFO) shows the rate of utilisation versus machining time, consumed to manufacture 40 workpieces. The average rate of utilisation of the machining centres is about 70 % (m/c #1) respectively 67 % (m/c #2) only. Due to the application of ETM the rate of utilisation is improved to better than 90 %. Thus, the total machining time to manufacture the given number of workpieces gets shortened.

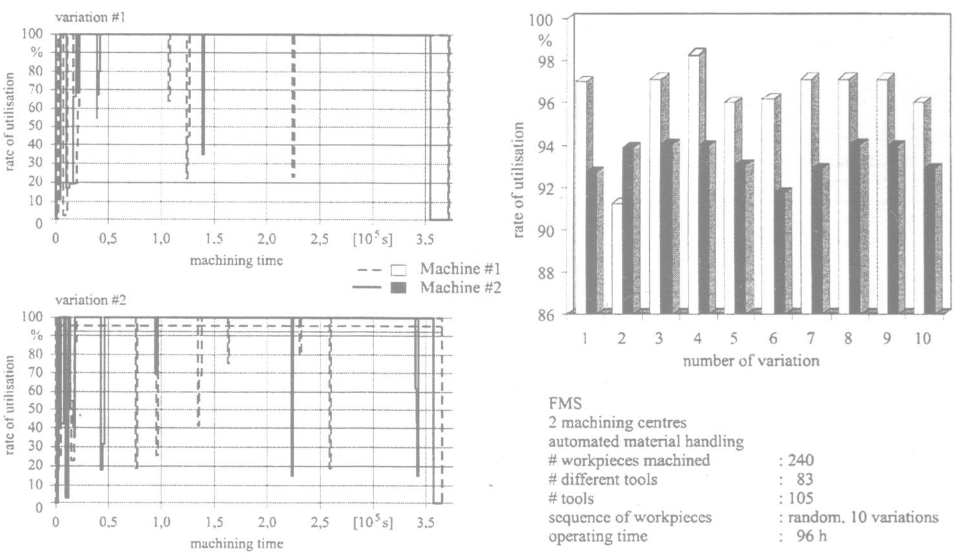


Fig. 4: Influence of sequence of workpieces onto the performance of the machining centres

As already mentioned, tardiness of tools will be notified, thus the planning department may rearrange the sequence of workpiece machining. The effects of rearrangement onto the performance of the machining centres is depicted in figure 4. Simulation was applied using ETM and the optimum number of tools. The sequence of about 240 workpieces of 7 different types was changed randomly. On the left hand side of the figure, e.g., the rate of utilisation of two variations versus machining time is given. In total, ten variations were examined, the results are shown in the bar chart on the right hand side. The rate of utilisation of machine #1 varies between 96% to 98%, whereas machine #2 is working to 91% to 94% capacity. Hence, once the number of tools is optimised and ETM is applied to schedule tools servicing the sequence of workpieces to machine can be changed without major impact onto the machine performance.

5. CONCLUSION

The Event-Oriented-Tool-Management system is presented and its performance is explained. The system is based on simulation and was developed to enable the planning and control of metal cutting tool and cutter preparation and servicing. Furthermore, by the interconnection with production planning and control features it enables the optimisation of the performance of manufacturing systems with respect to the tooling of the machines.

The capabilities of the system are demonstrated using the data of a FMS installed. It could be shown, that the number of tools in stock can be minimised and reduced as compared with the standard procedures to define the number of tools necessary to ensure the machining of a certain set of workpieces. Furthermore, the rate of utilisation of the machining centres under examination was improved.

Additionally, the ETM can be used to optimise the resources of tool preparation and servicing areas with special attention to the tool and cutter grinding machines. Based on simulation feasibility studies can be conducted and decision support for capital investment is given.

6. REFERENCES

1. Milberg J. and Ebner C., "Verfügbarkeit von Werkzeugmaschinen", VDI-Z 136, Nr. ½, 1994.
2. Maier P., "Störungsanalyse und Bestandsaufnahme an bestehenden Fertigungssystemen - Statistische Ausfallursachenanalyse-", in: Modellgestützte Fehlerfrüherkennung in der spanenden Fertigung. FQS-Schrift Nr. 95-01, 1991.
3. Simeonov S., "Simulation of Manufacturing Systems Using FACTOR/AIM", Proc. of the 7th European Simulation Symposium, ed. DalCin, Herzog, Bolch, Kaylan, pp.652-655, SCS, Erlangen, 1995.
4. Dörrsam V. and Furmans K., "Application Case Study of a Queuing Network Simulation Tool for Analysing and Optimising a Manufacturing System", Proc. of the 7th European Simulation Symposium, ed. DalCin, Herzog, Bolch, Kaylan, pp.652-655, SCS, Erlangen, 1995.
5. Baldi St. and Gmilkowsky P., "Modeling Distributed Production Control using an Integrated Multiagent an Descrete Event Simulation Approach", Proc. of the 7th European Simulation Symposium, ed. DalCin, Herzog, Bolch, Kaylan, pp. 610-614, SCS, Erlangen, 1995.
6. Dangelmaier W., "Dynamic Modelling of Production Processes as Basis for Distributed Objectoriented Production Planning and Control", Proc. of the 7th European Simulation Symposium, ed. DalCin, Herzog, Bolch, Kaylan, pp. 615-620, SCS, Erlangen, 1995.
7. Morton T.E. andPentico D.W., Heuristic scheduling systems, John Wiley & Son, Inc., 1993.
8. Hayes C.C., "A Manufacturing Process Planner for a Concurrent Engineering Environment", Proc. of the 1995 IEEE Intern. Symposium on Assembly and Task Planning, pp. 113-119, IEEE Computer Society Press, Los Alamitos, CA, 1995.
9. Krishnaswamy K.N., Raghavendra B.G. and Sampath D., "A Computerized Tool Planning and Scheduling System", Proc. of the 3rd Intern. Conf. on Computer Integrated Manufacturing, ed. Winsor, Sivakumar, Gay, vol. 1, pp. 802-811, World Scientific, Singapore 1995.
10. Pandey P.C. and Pal S., "Application of Neural Network in Selection Problems", Proc. of the 3rd Intern. Conf. on Computer Integrated Manufacturing, ed. Winsor, Sivakumar, Gay, vol. 1, pp. 812-819, World Scientific, Singapore 1995.
11. NN, "Reference Manual of Tool Data Management Softwaresystem", Walter Informationssysteme, Tübingen, 1994.
12. Petuelli G. and Müller U., "Planning and Control of Tool and Cutter Grinding in Manufacturing", Proc. of the 3rd Intern. Conf. on Computer Integrated Manufacturing, ed. Winsor, Sivakumar, Gay, vol. 1, pp. 795-801, World Scientific, Singapore, 1995.
13. Petuelli G. and Müller U., "Simulation of Rule-Based Metal Cutting Tool Resharpening", Proc. of the 7th European Simulation Symposium, ed. DalCin, Herzog, Bolch, Kaylan, pp. 674-678, SCS, Erlangen, 1995.
14. Petuelli G. and Müller U., "Optimization of the Performance of Manufacturing Systems by Integrated Tool Preparation Scheduling", Proc. of the 4th Intern. Conf. on Control, Automation, Robotics and Vision ,(ICARCV '96), Singapore, 1996.

NEIGHBORHOOD CONCEPT FOR EFFICIENT NESTING OF FLAT PATTERNS

S.K. Cheng and K.P. Rao
Department of Manufacturing Engineering & Engineering Management
City University of Hong Kong, Hong Kong

ABSTRACT

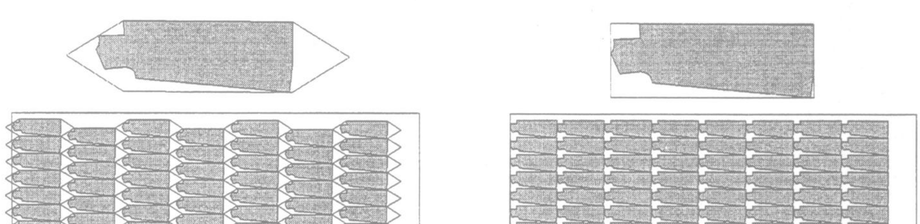
The traditional methods for solving the cutting stock problem, through clustering and nesting process, seem to be not intelligent enough to solve the problem efficiently due to numerous pattern positions that are possible. A *Compact Neighborhood* concept, that relates the number of neighbors and the sharing space between them, is introduced in this paper, and an algorithm is presented to provide quick and precise solutions. With the help of this algorithm, one can obtain an universal compact utilization of a specific pattern for bench-marking purpose. The clusters generated by this algorithm also provide geometrical hints for the subsequent nesting process for various types of layout.

KEYWORDS

Cutting Stock Problem, Pattern Nesting, Clustering, Compact Neighborhood

1. INTRODUCTION

The cutting stock problem is of interest to many industries. A typical intractable problem is to efficiently make a layout, the so called "Nesting", with highest stock utilization. Manual layout, performed even by an experienced marker, is both time-consuming and expensive because the nesting process normally involves many tedious operations as a consequence of the many possibilities which can be considered. With the growing development of computers, it has become very crucial to reduce lead times for this task so as to provide quick and precise solutions. Gilmore and Gomory [1] had initiated the research work by solving the rectangular cutting stock problem using linear programming. Adamowicz [2] used a heuristic approach which divides the problem into two sub-problems, called clustering and nesting. Clustering is to collect patterns that fit well together before nesting onto a given stock. Orientation of the patterns could be an additional constraint that has to be often satisfied. Traditionally, there are two basic techniques, called as "Hexagonal" and "Orthogonal" approximations, for generating nesting. In "Hexagonal Approximation", suggested by Dori and Ben-Bassat [3], a convex polygon is first approximated by another convex polygon but with smaller number of entities, and the new convex polygon is further converted into a hexagon that can be paved onto a stock, as shown in Fig. 1(a). Such approximation leads to very low accuracy, especially for highly irregular shaped patterns. In "Orthogonal Approximation", the nesting process is achieved by duplicating the patterns in an orthogonal way [4], as shown in Fig. 1(b). The yield achievable through this technique is also dependent on the shape of patterns.



(a) (b)

Fig. 1 : (a) Pattern enclosed in a regular hexagon and paved onto a rectangular stock, and (b) Pattern enclosed in a rectangle and paved onto a rectangular stock.

On the other hand, without a sophisticated definition of goal attainment, the nesting process can be regarded as a blind search of best layout that leads to global optimization. A cut-off heuristic was suggested to provide the stopping criterion and has been used with different techniques of artificial intelligence for solving the nesting problems [5, 6], such as simulated annealing and genetic algorithms. Generally, the termination of these algorithms follows the detection of improvements to the solution that are produced by successive iterations to satisfy a user-defined threshold limit. However, the determination of a suitable value of threshold limit is a tricky problem and so a high degree of automation, speed and reliability cannot be achieved while nesting a set of patterns.

2. COMPACT NEIGHBORHOOD AND UNIVERSAL COMPACT UTILIZATION

This study is aimed at improving the effectiveness of the conventional nesting processes by incorporating a concept of "Compact Neighborhood" which considers the relationship between the number of neighbors and the sharing space between them. Around a flat pattern its neighbors are sheared in order to constitute different orthogonal, sheared orthogonal or hexagonal close packing of equi-shaped patterns (unit cells) until the maximum possible density is reached. A more precise and useful measure of possible compactness of a specific pattern can hence be given as the packing density, which is the ratio of the area of an unit pattern to that of the unit enclosure, and can be termed as "Universal Compact Utilization". With this new definition, the maximum stock utilization for a specific pattern can be obtained and its validity is not limited to any special shape or dimensions of the stock sheet. As a result, the stopping criterion can be precisely fixed and the size of search space can be controlled by the use of different heuristic techniques. In addition, the crystallization directions that define the optimal ways for the duplication of cluster can be mathematically obtained during the formulation of a unit cell. These directions give useful information for the subsequent nesting process with the consideration of various types of layout.

2.1 Problem Representation and Definition

The proposed Compact Neighborhood Algorithm (CNA) tracks the characteristics of the evolving neighborhoods when the patterns are moved to form different arrangements, as shown schematically in Fig. 2(a)-(c), in the case of equi-circular patterns with radius r. It should be noted that the neighboring patterns are at closest possible locations around the pattern '*C*' which is regarded as the center of the neighborhood while the packing changes from orthogonal to hexagonal structure. A pattern is said to be the neighbor of a particular central pattern if there is at least one direct or indirect visible point from this pattern to the central one. Basically, what we want to study is the change in the neighborhood structure when the neighbors merge. In Fig. 2(a), the circles lie as an orthogonal packing and there are totally eight neighbors (N_n =8) to define the neighborhood. If the centers of all neighbors in contact are connected together, a square is formed which is termed as **Unit Cell**. By shearing the upper and lower layers, a sheared orthogonal lattice (N_n = 8) can be obtained, as shown in Fig. 2(b).

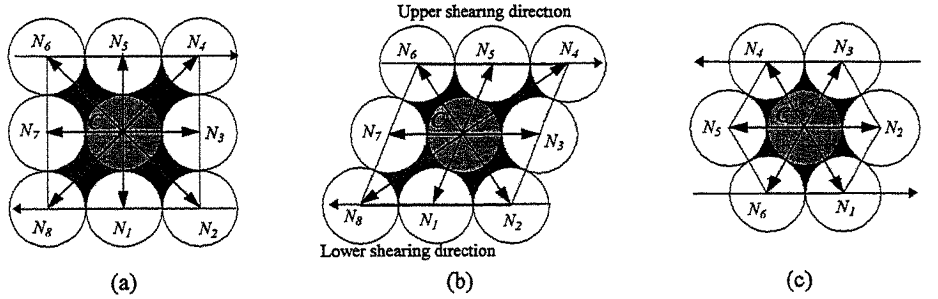


Fig. 2 : Typical neighborhood structures for circular patterns - (a) Formation of O.P. Unit Cell with N_n = 8 & A_u = 16r^2, (b) Shearing of layers leading to Sheared Orthogonal Packing (S.O.P. Unit Cell), and (c) Best compact structure with H.P. Unit Cell with N_n = 6 & A_u = 6$\sqrt{3}$ r^2.

As the shear displacement increases, the upper and lower neighbors tend to collapse. When adjacent rows have been displaced by a distance equal to the radius of the circle, the rows finally form a more tightly packed nesting as shown in Fig. 2(c). This is an hexagonal packing ($N_n = 6$) which is more dense and structurally more stable arrangement than the orthogonal nesting or the sheared orthogonal one. An universal measure of compactness of a neighborhood is the packing density, given by:

$$\text{Universal Compact Utilization } (UCU, \%) = \frac{N_{cd} \times A_p}{A_u} \times 100\% \qquad (1)$$

where A_p is the area of the pattern, A_u is the area of a unit cell and N_{cd} is the number of crystallization directions to construct the unit cell ($N_{cd} = 4$ in case of orthogonal or sheared orthogonal unit cell and $N_{cd} = 3$ in case of a hexagonal unit cell). With this definition, the compactness values of orthogonal and hexagonal nesting of a circle are 78.54% and 90.69% respectively, with the latter termed as *UCU* which is the maximum possible utilization if it is nested onto an in finite size stock with orientation constraint.

2.2 Proposed Technique for Developing Compact Neighborhood

Two dimensional patterns with entities of lines and arcs only are considered presently. A typical artwork pattern, as shown in Fig. 3(a), containing both concave and convex features, is selected to illustrate the working principle of **CNA**. The pattern contour is plotted with the help of a digitizer, as shown in Fig. 3(b), and has an area (A_p) of 74.44 sq. units. Four geometrical features of the central pattern '*C*' are first extracted or computed; the non-inclined rectangular enclosure with lower-left corner (P_{cl}) and upper-right corner (P_{ch}), centroid (P_{cc}) and the lowest-rightmost vertex (P_{cr}). The vertex P_{cr} on the central pattern '*C*' is regarded as the reference point of movement. The same nomenclature is applied to other patterns, for example, the centroid of a pattern '*E*' is point P_{ec}

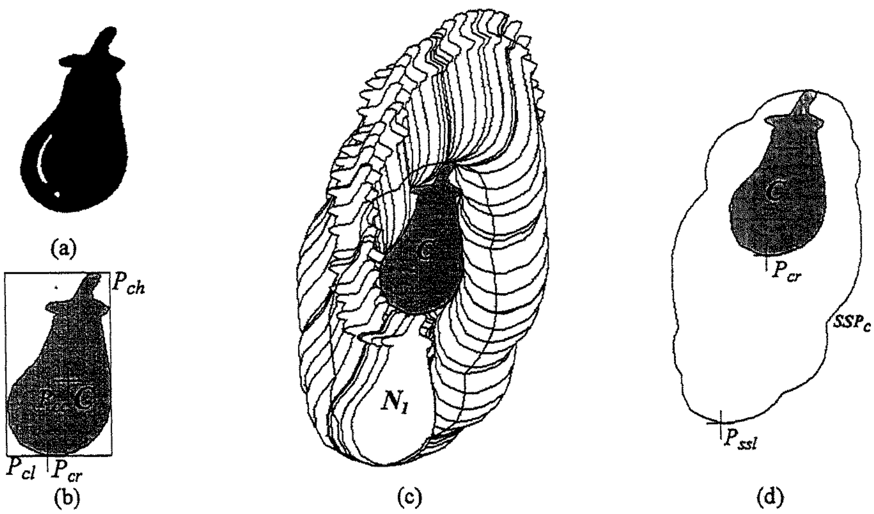


Fig. 3 : (a) Chosen flat pattern for demonstrating the working principle of CNA algorithm, (b) pattern contour with important points, (c) sliding steps involved in the analysis of the neighborhood, and (d) "Self-Sliding Path - SSP_c" of the central pattern.

The subsequent step is to duplicate pattern *C* itself. The first duplicated pattern is named as the "First Neighbor" - pattern N_1 - and it is then subject to the sliding process [7]. Fig. 3(c) summaries all possible positions obtained during the sliding process, and a "No-Fit-Polygon - NFP" is generated following the definition of Adamowicz [2]. NFP is used to guide the relative movement between two patterns with the consideration of no overlapping. In the example, NFP, as shown in Fig. 3(d), is plotted with respect to the reference point of pattern N_1 (P_{1r}). Since the sliding process is now operated by two similar patterns, i.e., patterns C and N_1, with the latter sliding along the stationary former, it is better to rename the No-Fit-Polygon as "Self-Sliding Path - SSP_c". It implies that it is a sliding path for an equi-pattern to move in touch with pattern C with respect to point P_{cr}.

With **path** SSP_c determined, one can now define the "First Crystallization Direction" - vector $\boldsymbol{CD_1}$ - joining the lowest point in path SSP_c (P_{sl}) with the reference point P_{cr} of the central pattern. A self-sliding path for pattern N_1 can also be generated and these two **paths** (SSP_c and SSP_1) are then subject to an union operation, which extracts the external non-overlapping features from the sliding paths. Fig. 4(a) shows the combined sliding path that defines the possible way for other neighbors to move around them. Line L_s, passing through points P_{cr} and P_{1r}, separates the combined sliding path into two halves which possess inverse mirror symmetry to each other. Geometrically, it is necessary to consider one of the sides only to define other crystallization directions. The right hand side of line L_s is selected for further study. In addition, only **path** $USSP_1$ which is from point P_{hc} to P_{he} is considered because, beyond this path, there is no neighborhood relationship for any pattern with pattern C. Now, vector $\boldsymbol{CD_1}$ and the centroid (P_{4c}) of "Fourth Neighbor - N_4" can be successfully defined.

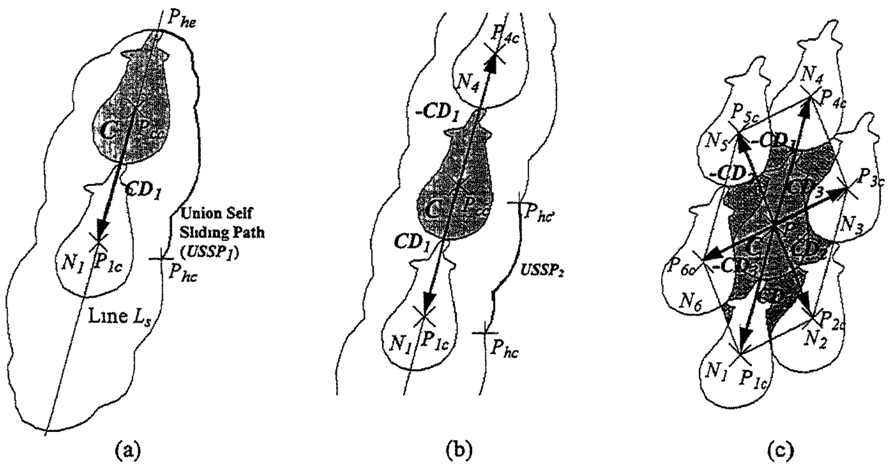
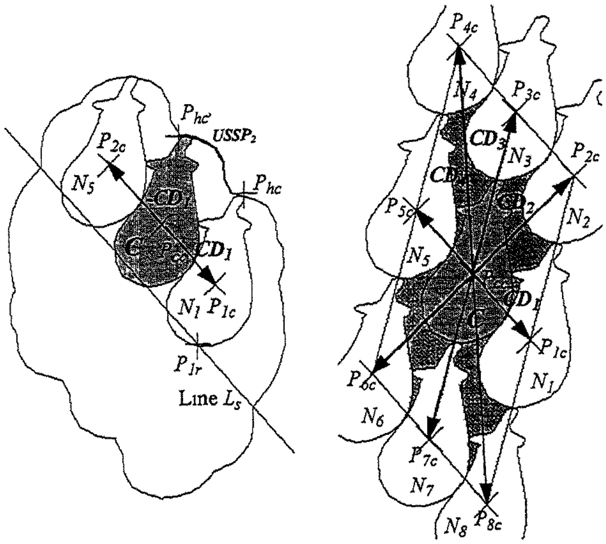


Fig. 4 : (a) Addition of "First Neighbor - N_1" and fixation of the First Crystallization Direction (CD_1), (b) Addition of "Fourth Neighbor - N_4", and (c) the generation of "Second and Third Crystallization Directions - CD_2 and CD_3" to form a H.P. unit cell with six neighbors.

It is only necessary to consider the efficient sliding path which enables the neighbor keep in contact with pattern C while it slides. So, a new combined **path** $USSP_2$ is generated, as shown in Fig. 4(b), by including one more self-sliding path around pattern N_4. There is an interesting point called "Hexagonal Crystallization Point (P_{hc})" which is the unique point for generating hexagonal packing, i.e. **H.P. unit cell**. After generating **path** $USSP_2$, the next step is to fix the "Second Crystallization Direction" - vector $\boldsymbol{CD_2}$ - by transforming the "Second Neighbor" - pattern N_2 - to the selected point P_{hc} in **path** $USSP_2$. The "Third Crystallization Direction" - vector $\boldsymbol{CD_3}$ - can be obtained by placing a third neighbor N_3 at the other extreme of $\boldsymbol{USSP_2}$, i.e. point $P_{hc'}$. A **H.P. unit cell** can now be constructed, as shown in Fig. 4(c), which leads to UCU of 75.48%. The same procedures are repeated by keeping vector $\boldsymbol{CD_1}$ constant but attempting other points on **path** $USSP_2$ to define new vector $\boldsymbol{CD_2}$ and then calculate the area of unit cell. However, these combinations lead to **O.P. or S.O.P.** unit cells with a total of eight neighbors. Therefore, the direction and magnitude of vector $\boldsymbol{CD_2}$ which leads to smallest area of unit cell, i.e., highest compactness, can be finally obtained with a fixed vector $\boldsymbol{CD_1}$.

After completing all points in **path** $USSP_2$, vector $\boldsymbol{CD_1}$ can then be modified by trying other points in **path** SSP_c following a counter-clockwise direction. For example, an intermediate position obtained is shown in Fig. 5(a). The line L_s is once again obtained by joining the reference points (bottom leftmost vertices) of the three patters shown, and vector $\boldsymbol{CD_1}$ is obtained by joining the centroids of these three patterns. Following the procedure mentioned earlier, one can obtain several **O.P.** or **S.O.P.** unit cells and one **H.P.** unit cell, with a sample of each indicated in Fig. 5(b) and (c) respectively. The *%UCU* for this particular **S.O.P. unit cell** is found to be 77.48%. From experience, a **H.P. unit cell** usually leads to the highest compactness but there is no guarantee. Repeating this technique for all other combination of vectors CD_1 and CD_2, the most compact neighborhood structure, shown in Fig. 5(c), with a *UCU* value of 83 07% can be obtained.

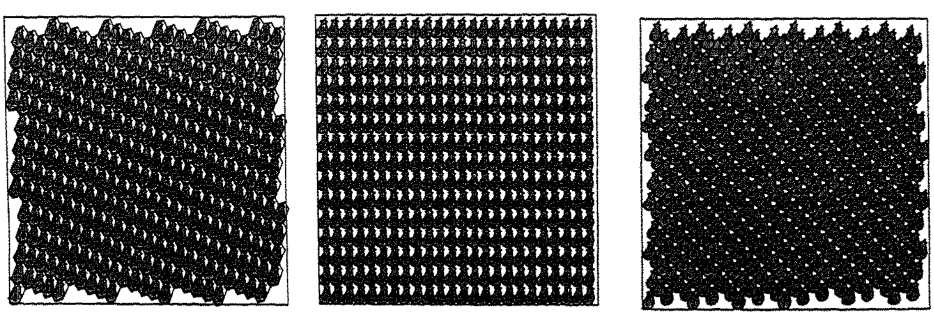


(a) (b) (c)

Fig. 5 : (a) Another possible packing structure obtained by changing the vector CD_1, (b) typical S.O.P. unit cell, and (c) optimal neighborhood structure with H.P. unit cell.

3. CASE STUDY

The efficiency of proposed Compact Neighborhood Algorithm (**CNA**) is compared with the traditional techniques ("Hexagonal Approximation" by Dori [3] and "Orthogonal Approximation" used by Nee [4]) in terms of stock utilization (*% SU*). Typical layouts generated are shown in Fig. 6, and the stock utilization achieved using various techniques for different stock sizes are compared in Fig. 7. As illustrated, the proposed **CNA** gives the best solution in all the cases, except when the width of the stock is very small, and converges with *UCU* as the stock size increases.



Hexagonal Approximation Orthogonal Approximation Generated by **CNA**

Fig. 6 : Possible layouts of the pattern on a stock of 200×200 units using different techniques.

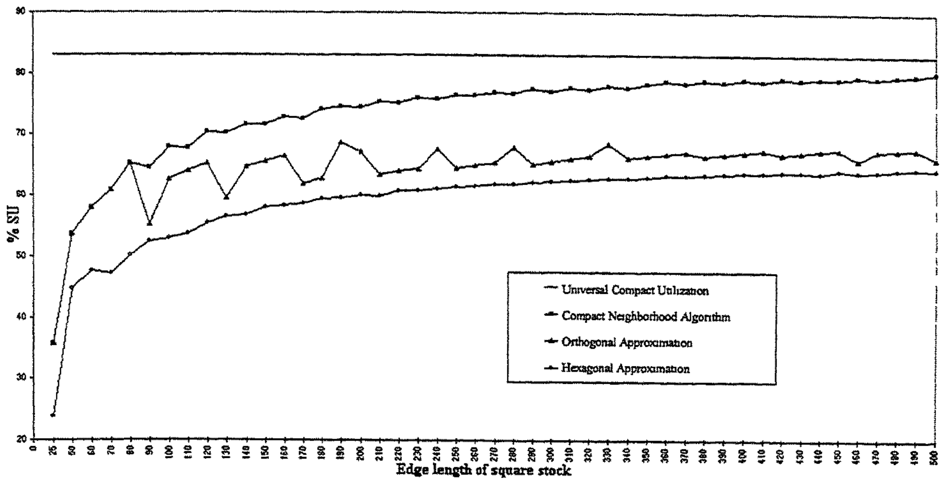


Fig. 7: Comparative utilization of stock with changing dimensions of the square stock.

4. CONCLUSIONS

A new strategy for quick and precise nesting of patterns composed of line segments and arcs has been successfully developed. The proposed Compact Neighborhood Algorithm detects the best layout among various possible solutions. Besides, it provides a stopping criterion that can be used in other nesting approaches to generate a near-optimal layout. A numerical value representing the best possible utilization, called "*Universal Compact Utilization*", is introduced, which serves the purpose of measuring the effectiveness of any nesting process or pattern layout.

5. ACKNOWLEDGMENT

The authors would like to acknowledge the financial support received from the City University of Hong Kong, through Strategic Grant No. 7000442.

6. REFERENCES

1. Gilmore, P.C. and Gomory, R.E., A Linear Programming Approach to the Cutting Stock Problem - Part 1, Vol.9, pp.724-746, Operations Research, 1961.
2. Adamowicz, M., The Optimum Two-dimensional Allocation of Irregular, Multiple-Connected Shapes With Linear, Logical and Geometric Constraints, Ph.D. thesis, Department of Electrical Engineering, New York University, 1969.
3. Dori, D. and Ben-Bassat, M., Efficient Nesting of Congruent Convex Figures, Vol.27, pp.228-235, Image Processing and Computer Vision, 1984.
4. Nee, A.Y.C., A Heuristic Algorithm for Optimum Layout of Metal Stamping Blanks, Vol.33, pp.317-320, Annals of the CIRP, 1984.
5. Jain, P., Fenyes, P. and Richter R., Optimal Blank Nesting Using Simulated Annealing, Vol.114, pp.160-165, Transactions of the ASME - Journal of Mechanical Design, 1992.
6. Jakobs. S., On Genetic Algorithms for the Packing of Polygons, Vol.88, pp.165-181, European Journal of Operational Research, 1996.
7. Cheng, S.K. and Rao, K.P., Quick and Precise Clustering of Arbitrarily Shaped Flat Patterns based on Stringy Effect, Vol.1, pp.69-72, Proceedings of the 20th International Conference on Computers & Industrial Engineering, Kyongju, Korea, October 1996.

STUDY ON FLEXIBLE AUTOMATIC INSPECTION SYSTEM

Y.Z. Wang, Y.H. Chen
Department of Mechanical Engineering
The University of Hong Kong
Hong Kong

ABSTRACT

This paper describes a flexible automatic inspection system(FAIS), which integrates computer aided design(CAD), computer vision, computer aided inspection planning(CAIP) and a coordinate measuring machine(CMM). The flexibility of the CMM has been greatly improved by making use of computer vision and computer aided inspection planning. It becomes possible to integrate a CMM into modern manufacturing system by using workpiece information obtained from CAD software directly. Working principles and the experiment results of the system are presented. The flexible automatic inspection system may be integrated into flexible manufacturing system(FMS) or computer integrated manufacturing system(CIMS) so that a comprehensive system capable of designing, manufacturing and inspecting can be achieved.

KEYWORDS

Automatic Inspection, CMM, Computer Vision, Computer Aided Inspection Planning, CAD

1. INTRODUCTION

The increasingly keen competition in the market greatly pushes forward the development of advanced manufacturing technology, which demands more flexibility and automation in mechanical manufacturing industry to suit the needs of single or small batch manufacturing. Therefore, more and more requirements such as precision, automation and flexibility etc. must be reflected in quality control especially in inspection technology.

Coordinate measuring machine(CMM) is a large precise and intelligent inspection machine which integrates optics, mechanics, numerical control and computer. Having advantages of large measuring space, high accuracy, universal and programmed etc., CMM is becoming the leading inspecting equipment in manufacturing especially in automatic manufacturing for the collection of quality data to achieve quality control. Fulfilling above requirements, a lot of efforts having been made to integrate modern technologies, such as computer aided inspection planning(CAIP) and computer vision, with CMM to increase its flexibility and level of automation.

Since 1980s researchers have paid more attention to CMM-based computer aided inspection planning[1~3], which enables coordinate measuring machine(CMM) to automatically generate inspection plan and collision-free probe path according to design information of workpiece. One research direction is to develop new computer aided design(CAD) software to suit CAD, computer aided manufacturing (CAM) and CAIP, which implies that a uniform product model will be used in computer integrated manufacturing system(CIMS). But as the development of CAD, CAM and CAIP the product model will be changed. The other is to make use of commercial CAD software. Linking commercial CAD software and CAIP, advice will be got on the uniform product model in CIMS.

Some research works have been done on integrating computer vision with CMM[4,5]. The main works focus on improving the flexibility and the level of automation of CMM by making use of computer vision. Computer vision not only enables CMM to have real-time recognition of workpiece type and its position and orientation information, but also to adjust inspection task automatically.

On the basis of related works above, a flexible automatic inspection system(FAIS) is proposed in this paper which integrates CAD, computer vision, CAIP and a CMM to achieve flexible and automatic inspection in advanced manufacturing environment, especially in flexible manufacturing system(FMS) and CIMS.

2. SYSTEM COMPOSITION AND WORKING PRINCIPLE

The flexible automatic inspection system consists of four sub-systems, they are: modeling sub-system, CMM-based CAIP sub-system, computer vision sub-system and CMM sub-system.

2.1 Modeling Sub-system

A widely used CAD software, AutoCAD, is used in the flexible automatic inspection system. The constructive solid geometry(CSG, a kind of 3-dimensional solid model commonly used in CAD software) is used to express the geometrical structure of workpiece[6]. When a workpiece is designed in AutoCAD, the modeling sub-system extracts its CSG-based geometry from drawing exchange file (DXF) files generated by AutoCAD. Then, the geometry is stored in the system data-storage and will be used for the creation of workpiece inspection plan and recognition templates.

2.2 CMM-based CAIP Sub-system

CMM-based CAIP sub-system works by rules and is divided into two parts: inspection process planning and probe path planning.

2.2.1 Inspection Process Planning

On the basis of the geometry and inspection information of a workpiece, inspection process of the workpiece and the details in it are created which includes classifying inspection items, accessibility analysis for inspected features, establishing workpiece coordinate system, arranging the inspected features, selection of probe and determination of inspected points for each inspected feature.

2.2.2 Probe Path Planning

According to the results of the inspection process planning, collision-free probe path is created which includes local path and global path. Local path is the probe trajectory when one feature is being inspected, including start point, off-set points, inspecting points and end point. Global path is the link of every local path to form the probe trajectory when one workpiece is being inspected. The main works in this part are selections of start point and end point, off-set points generation, collision testing and the creation of retreating probe path. The collision-free probe path is stored in the system data-storage and will be used to direct the inspecting operation of CMM.

2.3 Computer Vision Sub-system

The first task of computer vision sub-system is to create templates of multiple orientation run length coding(MORLC) of designed workpiece according to its geometry. The templates are used to recognize inspected workpiece. The second task is to recognize, position and orient the inspected workpiece on CMM table.

2.3.1 Multiple Orientation Run Length Coding(MORLC)

The multiple orientation run length coding is a coding method of workpiece binary images[6]. The recognition of 3-dimensional workpiece becomes more easy by making use of it. The MORLC is based on orientation run length coding(ORLC, its details can be seen in [5]) and overcomes its disadvantage, templates are confused sometimes, by making use of the coding of multiple binary images of workpiece taken from multiple fixed positions relative to the camera of computer vision sub-system. The coding results are named MORLC templates:

$$MORLC = \{M_0ORLC, M_1ORLC, \cdots, M_kORLC, \cdots, M_nORLC\} \tag{1}$$

where k is the number of workpiece position relative to the camera, $0 \le k \le n$. MORLC templates are divided into MORLC coordinate templates, MORLC length templates and MORLC byte templates. They can be expressed as follows, respectively:

$$MPM = \{M_0PM, M_1PM, \cdots, M_kPM, \cdots, M_nPM\} \tag{2}$$

$$MLM = \{M_0LM, M_1PM, \cdots, M_kLM, \cdots, M_nLM\} \tag{3}$$

$$MBM_{m\times n} = \{M_0BM, M_1BM, \cdots, M_kBM, \cdots, M_nBM\} \tag{4}$$

2.3.2 Creating MORLC Templates According to Workpiece Geometry

According to workpiece geometry, imaging process is simulated to draw the binary images of workpiece on the monitor of computer vision sub-system when it is on different fixed positions relative to the camera. Then workpiece MORLC templates are calculated and stored in the system data-storage. When simulating imaging process, three conditions are considered which are the illumination, surface reflection of workpiece and the parallax of optical system.

2.3.3 Recognizing Workpiece by MORLC Templates (MORLC Templates Matching)

The procedure in recognizing workpiece by use of MORLC templates is actually the matching of stored templates and templates of being recognized workpiece. A workpiece can be recognized by matching its coordinate templates or its length templates or its byte templates. When MORLC coordinate templates are used, the matching operation will be finished when the squared sum of the difference of corresponding terms between one of the MORLC coordinate templates in the system data-storage and the MORLC coordinate template of the workpiece being recognized have the minimum, that is:

$$\phi_j = \sum_{k=1}^{n} (M_kPM_{workpiece} - M_kPM_j)^2 \tag{5}$$

$$MINIMUM\{\phi_1, \phi_2, \cdots, \phi_j, \cdots, \phi_m\} \tag{6}$$

where k is the number of workpiece position relative to the camera, $0 \le k \le n$; j is the index number of MORLC coordinate templates in the system data-storage, $0 \le j \le m$. The matching principle of MORLC length templates is the same as above. The matching operation of MORLC byte templates is a logical operation. The matching operation will be finished when the result of 'and' operation between one of the MORLC byte templates in the system data-storage and the MORLC byte template of the workpiece being recognized is equal to that MORLC byte templates.

2.3.4 Positioning and Orienting of Workpiece

The positioning and orienting space of workpiece on CMM table is 2-dimensional. To achieve quick and flexible positioning and orienting, the centroid(X, Y) of workpiece binary image is used to specify its position, that is:

$$X = \sum_{i=o}^{n}\sum_{j=0}^{m} f(x,y)\cdot x \Big/ \sum_{i=0}^{n}\sum_{j=0}^{m} f(x,y) \tag{7}$$

$$Y = \sum_{i=o}^{n}\sum_{j=0}^{m} f(x,y)\cdot y \Big/ \sum_{i=0}^{n}\sum_{j=0}^{m} f(x,y) \tag{8}$$

where i, j are respectively the row and column index number in a digital image, $0 \le i \le n$, $0 \le j \le m$; f(x,y) is the density function of a digital image. Workpiece orientation angle is presented by the angle between the major axis of its binary image and the X axis of the visual coordinate system. The major axis passes the centroid(X,Y) of workpiece binary image and corresponds to the minimum of inertia of it[5]. That is:

$$\theta = \frac{1}{2}\arctan\left(\frac{2I_{11}}{I_{20} - I_{02}}\right) \tag{9}$$

where I_{11}, I_{20} and I_{02} are second-class moment of inertia.

2.4 System Working Process

The system working process is illustrated in Figure 1. Recognition templates and original probe path of designed workpiece are created and stored in system data-storage in advance. When inspected workpiece is put on CMM table, its MORLC templates are generated by the computer vision sub-system which are used to match the MORLC templates of designed workpiece stored in the system data-storage. According to the results of matching, the planned collision-free probe path of this inspected workpiece are fetched from the system data-storage and are rectified according to the position and orientation of the inspected workpiece on CMM table. Then the rectified probe path is transmitted to CMM sub-system. The automatic inspection of the workpiece is accomplished by CMM finally.

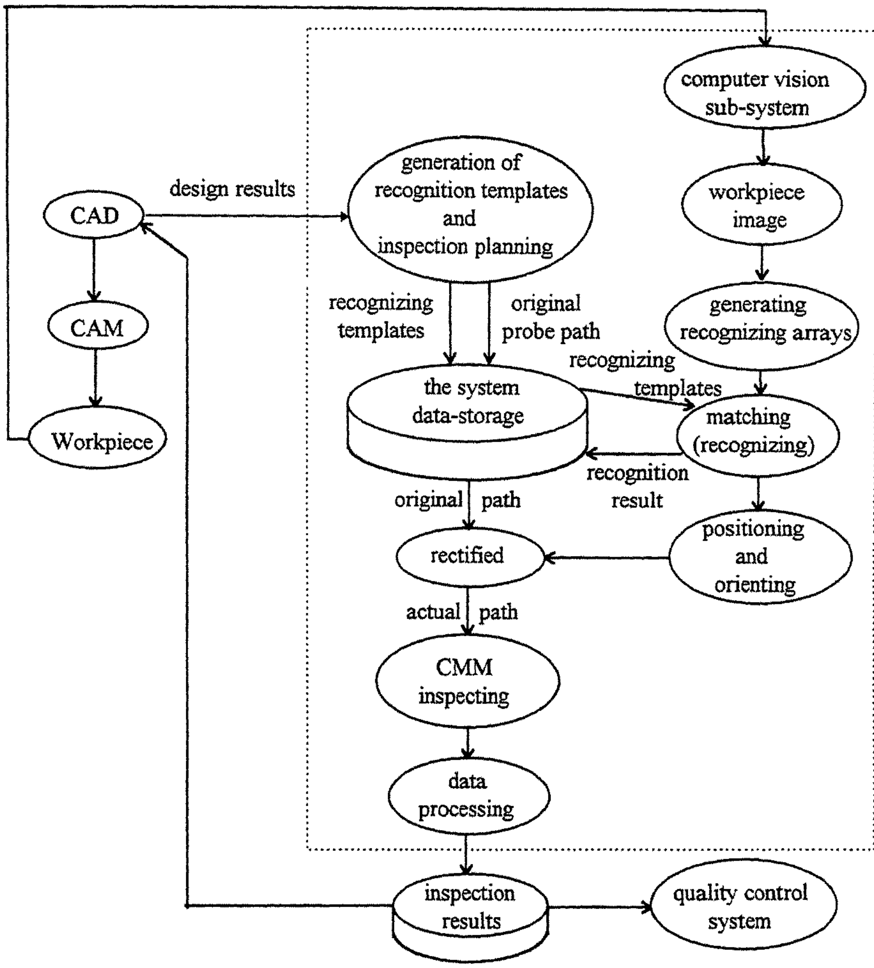


Figure 1 system working process diagram

3. EXPERIMENTAL CASES

A series of experiments are accomplished to test the functions of the flexible automatic inspection system. These experiments includes: To pick up the workpiece CSG-based geometry from DXF files; To create the MORLC templates from workpiece geometry; To recognize, position and orient the inspected workpiece by computer vision sub-system; To generate the inspection plan of workpiece.; To rectify the original inspection path into actual path; To inspect workpiece automatically, etc. The inspecting experiments were performed on the OPTON UMC 850 CMM and the other sub-system run in a IBM PC. The resolution of computer vision sub-system, which consists of a CCD camera and a image processing card, is 512H×512V and 256 grey levels. Some inspection results of one workpiece as shown in Figure 2 are given in Table 1.

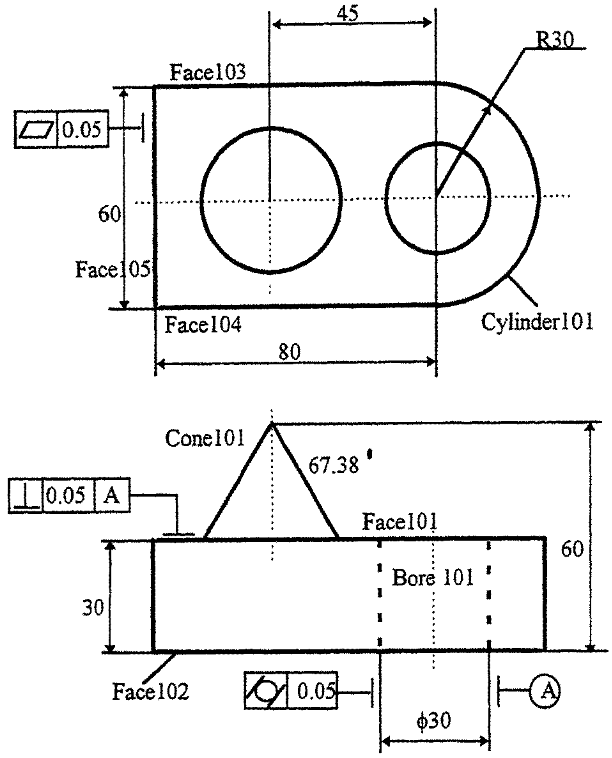


Figure 2 Drawing of a typical workpiece

The workpiece was inspected when it was at different positions and orientations on CMM table. From Table 1, it can be seen that the inspecting results are insensitive to the position and orientation of workpiece on CMM table.

Table 1 Some inspecting results when the workpiece was at different positions and orientations

inspected item	position and orientation1	position and orientation2	position and orientation3	position and orientation4	position and orientation5	position and orientation6
diameter1	30.158	30.159	30.160	30.160	30.160	30.160
cylindricity1	0.012	0.015	0.015	0.012	0.012	0.013
parallelism1	0.098	0.095	0.097	0.097	0.097	0.096
perpendicularity1	0.222	0.219	0.220	0.216	0.216	0.216
flatness1	0.001	0.001	0.001	0.001	0.001	0.001
distance1	60.011	60.009	60.010	60.010	60.011	60.010
distance2	79.870	79.873	79.872	79.871	79.872	79.872
distance3	45.432	45.432	45.433	45.435	45.433	45.433

From the above inspecting results, we are sure that all sub-systems of the flexible automatic inspection system work well and the workpiece inspection accomplished by the system is reliable.

4. CONCLUSION

The research work in this paper shows that it is a feasible and effective method to combine CMM with the technologies of CAD, computer vision and computer aided inspection planning, so as to increase the flexibility and level of automation of CMM. The flexible automatic inspection system formed in this way can be integrated into flexible manufacturing system(FMS) or computer integrated manufacturing system(CIMS) so that a comprehensive system capable of designing, manufacturing and inspecting can be achieved.

5. REFERENCE

1. Evershein, W. et al., "Automatic generation of parts programs for CNC-coordinate measuring machines linked to CAD/CAM system", Annals of the CIRP, Vol. 35, No. 1, pp.341-345, 1986
2. Yau, H.T. et al., "Path planning for automated dimensional inspection using coordinate measuring machines", Proceeding of the 1991 IEEE International Conference on Robotics and Automation, Sacramento, California, pp. 1934-1939, Apr. 1991
3. Medland, A.J. et al., "The integration of coordinate measuring machines within a design and manufacturing environment" , Journal of Engineering Manufacturing, Vol. 207, No. 132, pp. 91-98, 1993
4. Takeuchi, Y., Shimizu, H., Mukai, I., "Automatic measurement of 3-dimensional coordinate measuring machine by means of CAD and image data", Annals of the CIRP, Vol. 39, No. 1, pp. 565-568, 1990
5. Wang, X.J. "A research on flexible and automatic measurement in FMS and CIMS by making use of CNC-CMM and machine vision", a dissertation for Ph.D., Tianjin University, 1993
6. Wang, Y.Z. "A study on robotistic flexible automatic coordinate measuring system", a dissertation for Ph.D., Tianjin University, 1995

AN OBJECT ORIENTED METHODOLOGY FOR OPERATION & MACHINE SELECTION OF FEATURE BASED PROCESS PLANNING SYSTEM

S.V.Bhaskara Reddy[1], N. Bala Krishna[1], M.S. Shunmugam[1] and T.T. Narendran[2]

[1] Manufacturing Engineering Section
Department of Mechanical Engineering.

[2] Industrial Engineering & Management
Humanities & Social Sciences Department.

Indian Institute of Technology
Madras - 600 036., India.

ABSTRACT

The present paper stresses the need for a feature based process planning system that is capable of representing both technological and geometric information in an effective way. The usefulness of Object Oriented Paradigm (OOP) in the selection of operation and machine is explained. The application of OOP enables the present system more flexible in terms of future modifications.

KEY WORDS

Computer Aided Process Planning, Form Feature, Object Oriented Methodology.

1. INTRODUCTION

Process Planning is an art of translating part design information from an engineering drawing into a feasible technological sequence of manufacturing instructions to produce a finished product. Of the two important approaches in Computer Aided Process Planning (CAPP), namely Variant and Generative, the former suffers from drawbacks like laborious preparatory phase, inflexibility and the necessity for frequent updating of process plans, whereas the Generative approach uses decision logic, part definition data, heuristics etc. to automatically generate a process plan [1, 2].

In any generative CAPP system, Part definition data and the input format play an important role in deciding the capability of the system and its ease of use For making the planning decisions like operation selection, sequencing etc., the information contained in a CAD model has to be recognized in the form of features [3]. Some of the most elusive problems in the development of a truly generative CAPP system are as follows [4]:

- The inability of Feature recognition techniques in the current CAD systems to generate output with much useful technological information (tolerances, surface finish etc.) that are essential in subsequent activities like process planning.
- The difficulty in acquiring, organizing and representing the knowledge required to implement the large number of functions (Selection of operations, tools, machines etc.) and their overlap involved in the process planning. Thus many existing CAPP systems developed in traditional programming languages are too complex to manage in terms of future modifications.

To obviate the above obstacles, a Feature based model with an Object Oriented methodology is a viable alternative for representing and storing the geometric and technological information of each product feature. With this new paradigm, manipulation of these features becomes easy to assist the task of planning decisions such as type of cut (rough or finish), manufacturability and accessibility (by tool) etc. The main goals of Object Oriented Design (OOD) are Reusability and Maintainability. Process Planning being a very complex system with interaction of many classes taking place simultaneously, it needs to be developed in many stages. Designing the system keeping the anticipated changes in mind leads to a better design [5,6]. In the present work Maintainability is used as the main criterion in designing the system.

The present paper reports some of the ongoing efforts to develop a Feature Based Process Planning System (FBPPS) for rotational components. Thrust is given on the implementation of planning functions viz. selection of operations and machines by making best use of the capabilities of OO

Paradigm such as encapsulation, inheritance and polymorphism. This work is being done using Borland C++ language on an IBM compatible PC/AT 486.

2 METHODOLOGY

Object Oriented Approach requires the identification of a set of classes from problem domain and express the working of system as the interaction between the instances of these classes. Each class models the behavior of an object in the real world. The system works by passing messages between objects to perform certain functions.[7]. The different base classes identified in the present work are Process Plan, Part, Form Feature, Operation, Machine Tool, Cutting Tool. Fig. 1 represents the hierarchy of Process Plan which includes the messages passed between the objects instantiated from different classes. In the present system, the definition of an object representing a rotational part contains different data members and functions. The data members include part's name, number, material specifications, features with geometric & technological information such as dimensions, tolerances, surface finish etc. The various functions include Select operations, tools, machines and Sequence of operations etc.

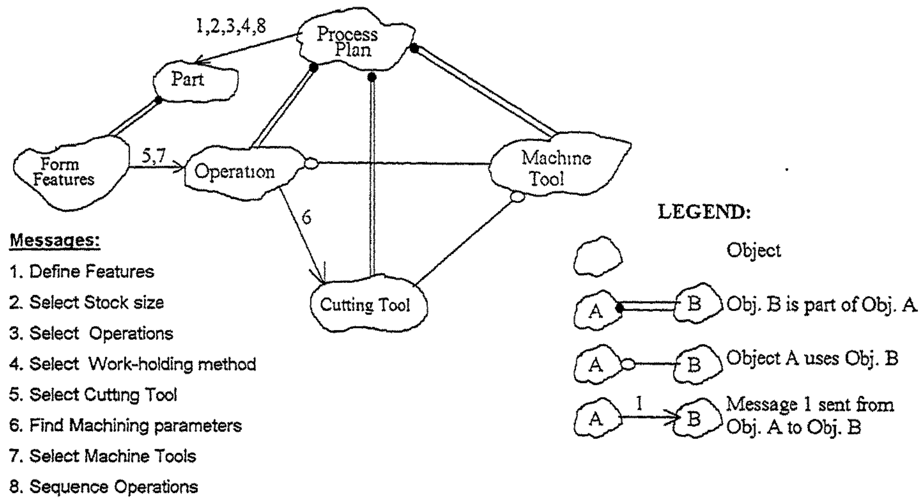


Fig. 1 : Process Plan Class Hierarchy.

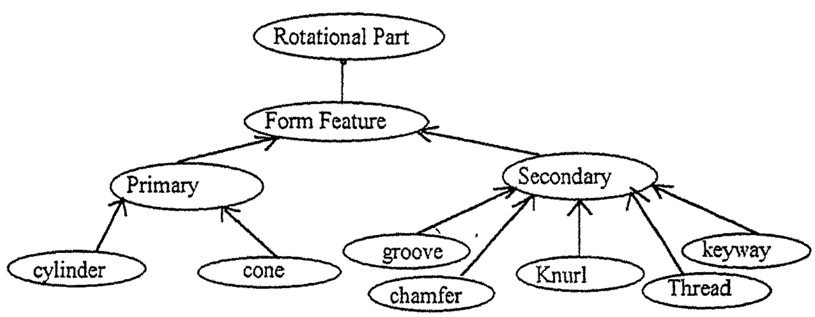


Fig. 2 : Hierarchical Structure of Form Features.

A machined part is represented as an assembly of form features like cylinder, cone, hole, thread, chamfer, fillet etc. A base class 'form_feature' is defined with all the common data members such as: reference length, type & subtype of feature (external/internal & primary/secondary), basic tolerances and surface finish, material specification. etc. Taking advantage of Inheritance in the present methodology, these data members are inherited to the derived classes like cylinder, cone etc. It is to be noted that a cylindrical hole is considered as an internal cylinder and likewise other holes. Hierarchical structure of form features with different primary and secondary features is shown in Fig. 2.

3. IMPLEMENTATION

3.1 Part Definition Data - Input Format

Cylinder and Cone are the two primary form features identified for the rotational parts. The attributes for cylinder and Cone are very similar except that the former has a single diameter and the latter has major and minor diameters. Secondary form features such as chamfer, groove etc. are often found on primary features and have to be associated with their primary features. Fig. 3 shows an example rotational part having different primary and secondary features.

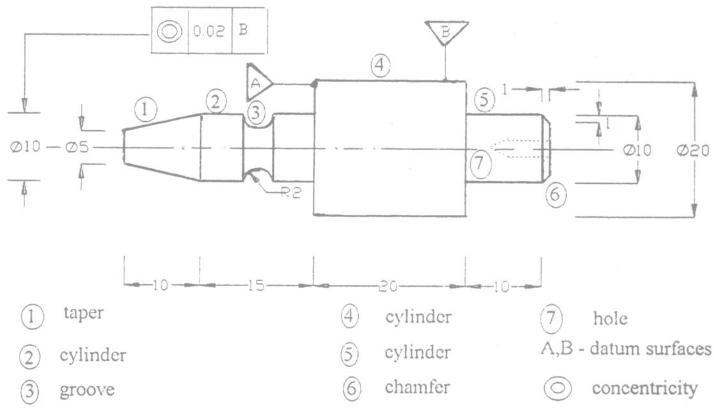


Fig. 3 : An Example Part.

To obtain the part definition data, a user-friendly "Feature Editor" as shown in fig. 4 is written in Visual Basic that allows the user to define and alter the feature attributes, such as dimensions, reference positions, geometrical tolerances, surface finish etc. This information is stored in a data file which is later read by the system to instantiate the system objects.

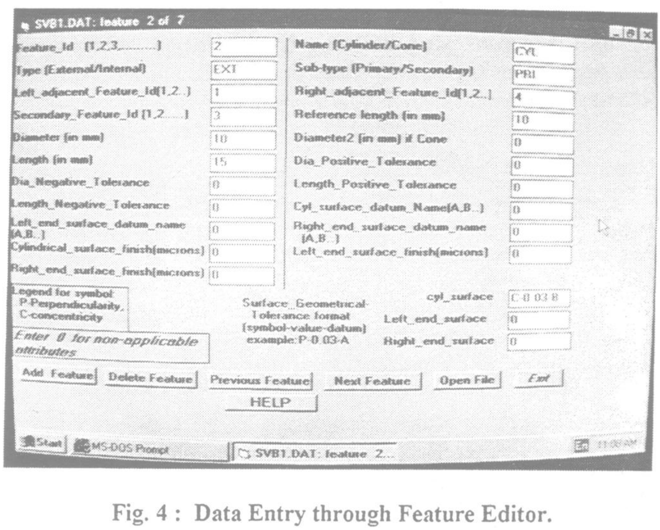


Fig. 4 : Data Entry through Feature Editor.

For the example part, sec_feature attribute gives a list of secondary features that are part of the primary feature under consideration. While dealing with rotational parts, it is quite obvious that any primary feature (i.e. cylinder or cone) would be bound by a maximum of 3 surfaces i.e. one cylindrical, two end surfaces (Left or Right) and 2 primary features (Left adjacent & Right adjacent). For each feature's surface, attributes are given for geometric tolerance specification, surface datum names and surface finish requirements.

While preparing the process plan, the first task would be to select the operations or processes which are required to manufacture the part and later select the machine tool based on the cutting tool and optimum cutting conditions in an iterative manner.

3.2 Operation Selection

Aim of the Operation selection is to determine what operations possess the capability to manufacture the features that comprise the finished part. Operation selection is the primary step in process planning The selection is done by matching type of feature and its specifications such as geometry, surface finish and tolerance requirements to the appropriate operation or series of operations for each feature. Operation class hierarchy as shown in Fig.5 is used to define shape generation capabilities of each operation and the Form feature class hierarchy contains information on geometry and technological requirements of each form feature.

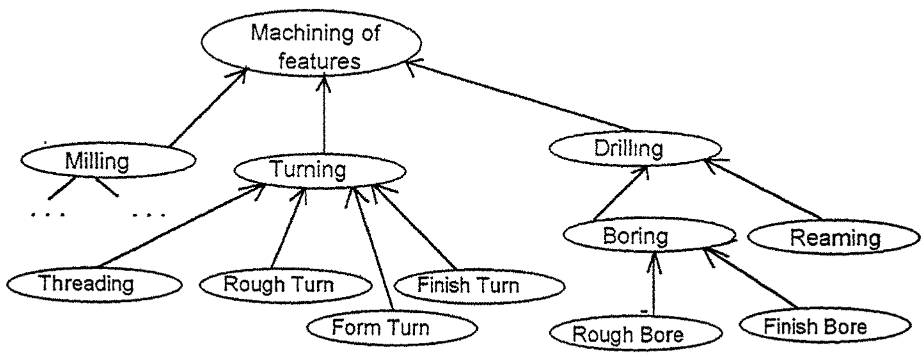


Fig. 5 : Operation Class Hierarchy.

Operation selection is invoked by passing the message 'select_operation( )' to the form-features of the part. Implementation of this function will not be the same for all features as the production rules differs from feature to feature. With in each form-feature class, the select_operation( ) function defines the procedure for selection of an operation or sequence of operations to produce that feature. The use of polymorphism permits a client object (part) to issue the same message, select_operation( ), to any feature, prompting it to perform the appropriate operation selection Also, the concept of Polymorphism facilitates the expandability of the system for new derived feature types. For example, to add a new form feature (eg. knurl) to the process planning system, a new derived class (i.e. knurl) can be derived from the form_feature class and the behaviors of the new class implemented

In literature, different methods are available for translating the feature specifications to required operations which include, Direct Association [8,9], Decision tables & trees [1] and Production rules [10].In direct association, each feature is specified by an operation or series of operations to produce it. By this method, selection is done without considering the specific attributes of the feature. Though easy to implement, it lacks flexibility in considering alternative operations for machining a feature. The drawback of direct association is overcome in other approaches where process knowledge of each operation is represented using decision tables, trees or production rules. However, the efficiency of these approaches is lowered as the system must go through all the knowledge available in tree or table or rule-base (what ever the case may be) to determine the possible alternative operations.

In the present system, selection is implemented by combination of direct association and production rules. In each feature class, a set of 'possible operations' to be considered for producing that specific feature are defined. If the system finds only a single operation associated with a feature, then it accepts this operation as the "Selected operation", and no further processing is done. If multiple operations are listed, selection of the appropriate alternative is done by examining the rules associated with those operations listed. This method combines the advantages of direct association (i.e. simplicity and efficiency) with the ability to consider alternative operations. It also limits the rule-base search to only those rules associated with the operations listed.

3.3 Example

Considering a cylindrical hole, the *possible_operations* data member lists the alternative operations that the system should consider in selection.

possible_operations = {{drill},{drill,rough bore},{drill,ream},{drill,rough bore,finish bore},
{drill, rough bore, finish bore, ream}}

There are 5 alternative operation series. Members of a particular series define the precedence that is required for the production of associated feature. Each operation is represented by an object which expresses the geometrical and technological capabilities such as Max. Diameter, Min. Diameter, Min. Surface finish and tolerances that can be achieved etc. of the operation. In a similar way, the data members of the object corresponding to form feature class would represent the design requirements for each feature. The system would check the alternative operations from the possible_operations list and find the operations that match the feature's requirements.

A typical production rule that is being used for selection of operation for a hole is as follows [11]:

If (Dia > *Min. Dia*)
and (Dia < *Max. Dia*)
and (Surface Finish > *Min. Surface Finish*)
and (Positive Dia Tolerance > *Min. Positive Dia Tolerance*)
and (Positive Length Tolerance > *Min. positive Length Tolerance*)
then (Selected_Operation = *Operation*)

The italicized parameters represent the operation attributes and non-italicized ones represent the feature requirements of the hole class.

The selection process starts by considering the first operation in the series, if no match is found the next operation in the series is considered. This procedure is continued until either a match is found or end of the list is encountered. When the system selects an operation from a series say 'finish bore' in the fourth series of possible operations list mentioned earlier, it implies that all operations that precede this namely drill, rough bore are included in the list of 'required operations' for that feature.

This approach eliminates the need for a central knowledge-base as it locates the operation selection rules for a feature with in the feature class itself. This facilitates the customization of rules to suit each feature types. The selection process does not require an inference engine that is essential in knowledge-based process planning systems. If one needs to add or modify rules, the present system facilitates easy modification when compared to conventional rule-based systems.

4. MACHINE SELECTION

Before machine selection, cutting tools are selected for a particular operation by accessing the cutting tool data base using production rules [12]. Then machining parameters such as speed, feed, depth of cut are calculated for different operations based on direct search algorithm [13]. In the present work only Lathe and Milling machines are considered. Machine Selection involves finding a match between capabilities of a machine and the requirements of the part and operation. Each machine in the shop floor is an instance of an object from machine class. The data members of this class are: machine type, operation type, work table size, available spindle horse power, feed and speed range, tolerance capability etc. Information regarding the dimensions of the part is available to each feature object by means of inheritance. A typical production rule for the selection of machine tool:

If (operation type = "turning")
and (part length < max. part length)
and (part diameter < max. part diameter)
and (required feed < max. feed)
and (required feed > min. feed)
and (required speed < Maximum speed)
and (required speed > Minimum speed)
then machine = " Lathe 01".

When there are alternative machines available for performing an operation, Machine selection is based on minimum cost criteria.

5 CONCLUDING REMARKS

The present work demonstrates the usefulness of OOP in the design of FBPPS. The present work is more flexible in terms of easy modification to the system. The implementation of Planning Functions by making use of Inheritance and Polymorphism, the two powerful features of OOP eliminate the need for a separate Knowledge-base and Inference Engine, instead it relies on the ability of the objects for storing the knowledge. Different modules of FBPPS are under development. The integration of these modules is being carried out.

6. REFERENCES

1. Chang T.C. and Wysk R.A - An Introduction to Automated Process Planning systems, Prentice-Hall Inc., 1985.
2. Mukasa E Ssemakula and Ramesh M. Rangachar.,"The Prospects of Process Sequence Optimisation in CAPP Systems", Computers & Industrial Engg., V 16, n 1, pp 161-170, 1989.
3. Rong-Kwei Li and David D. Bedworth, "A Frame Work for the Integration of Computer Aided Design and Computer Aided Process Planning", Computers & Industrial Engg., V 14, n 4, pp 395-413, 1988.
4. A.H. Juri, A. Saia and A. De Pennington, "Reasoning machining operations using feature-based models", Int. J. Prod. Res., V 28, n 1, pp 153-171, 1990.
5. Emdad. H. Khan, Mansoor Al-A'ali and Moheb R. Girgis, "Object-Oriented Programming for Structured Procedural Programmers", Computer, pp48-57, 1995.
6. Booch G., Object -Oriented Design with Application, The Benjamin/Cummings Pub. Co, 1991.
7. Usher J M, "An Object -oriented Approach to product modeling for manufacturing systems", Computer and Industrial Engg., 25(1-4), pp. 557-560, 1993.
8. Korde, U.P., Bora, B.C., Stelson, K.A., and D.R.Riley, "Computer aided process planning for turned parts using Fundamental and Heuristic principles", Trans. Of ASME J. Of Engg for Industry, 114, pp. 31-40, Feb. 1992.
9. Hinduja, S., and Huang, H, "OP-PLAN: An automated operation planning system for turned components", Proc. Instn. Mech. Engrs., 203, pp. 145-158, 1989.
10. Yeo, S.H., Wong, Y.S., and M.Rahman, "Integrated Knowledge-based machining system for rotational parts", Int. J. Prod. Res., 29(7), pp. 1325-1339, 1991.
11. H P (Ben) Wang , R A Wysk, "A knowledge-based approach for automated process planning", Int. J. Prod. Res., V 26, n 6, pp 999-1014, 1988.
12. Chen S.J., Hinduja, S. and G. Barrow, "Automatic Tool Selection for Rough Turning Operations", Int. J. Mac. Tools. Manufact., V 29, n 4, pp 535-553, 1989.
13. Aresecularatne, J.A., Hinduja, S. and G. Barrow, "Optimum Cutting Conditions for Turned Components", Proc. of Inst. of Mech. Engrs., V 206, n B1, pp 15-31, 1992.

DECISION-MAKING SYSTEM FOR PROCESS PLANNING USING A CONSTRAINT PROGRAMMING APPROACH

L. M. Galantucci, R. Spina, L. Tricarico

ABSTRACT

In this paper the authors use a constraint programming approach to design a decision-making system for process planning activities; the system is integrated with a CAD/CAPP interface, that describes the part in term of manufacturing features, and with a resources interface, that describes the capability of the manufacturing system; CAD data and production information are organised in two object-oriented databases. Using constraint programming, the technological rules and the problem objective functions have been formulated in form of constraint expressions; the decision-making program retrieves objects from the two databases and applies search algorithms, as backtracking and tree traversal, to evaluate all the possible solution goals of the constraint expressions. The system has been implemented to select the resources (processes, cutting tools, cutting parameters) for the machining of prismatic workpieces, that have simple holes as form features with or without precision requirements.

KEYWORD

CAD/CAPP, Decision-making system, Constraint programming.

1. INTRODUCTION

Computer Aided systems in the last two decades have improved product quality and manufacturing production capability, realising in the same time a reduction of design and production costs; the possibility to integrate the design and the process planning phases represents an important way to reduce the time to market and the costs of a mechanical component /1/. The phases of process selection and process parameters choice need a great effort in a manufacturing environment and have a significant influence on the final product quality; usually this task is done by the process planner, he prepares the necessary work instructions to produce a mechanical component, starting from a raw material or a semi-finish product /2/. This activity can be defined as a decision-making process, with a lot of factors that influence final process solution; it addresses towards processes and resources selection (machines tool, cutting tools, etc.) with the aim to fulfil the product characteristics (form, precision and technological features) and the capability of the manufacturing system. All these reasons highlight the importance to create a decision making system able to support process planner; one of the main problems is the knowledge representation used to define the decision making schema. Rule-based systems are currently the most used /1,2/; they are relatively simple but they have several disadvantages, as the difficult system implementation when it is necessary to handle a wider application area. Recently new approaches have been proposed to develop manufacturing decision support systems such as neural networks, constraint networks, constraint programming /3/.

The constraint programming can analyse and solve problems which belong to the category of the Constraint Satisfaction Problem (CSP) /4/. In general a CSP is represented with a set of variables, their variability field and a list of constraints; the solution search consists in the determination of the variable values which satisfy a set of imposed limitations. This CSP category is sometimes defined as a finite resources problem, because its goal is to find an optimal distribution of the resources (values of the variable which belong to the variability fields and satisfy the constraints). In the more complex case in which the CSP has also an objective function, the CSP aim is the optimisation of this function, respecting the operative limitations tied to the type of the problem. The constraint programming technique represents an alternative and more efficient way to solve a CSP, because it allows to save specific knowledge of the problem without the necessity to separate the problem and its model of representation. Furthermore, this technique is characterised by some features which accelerate the solution search; these are the constraint propagation (that allows the analysis domain reduction), and solution search algorithms, as the Tree Traversal and the Backtracking.

In this paper the concepts of constraint programming have been used to formalise a system architecture schema for a decision making system; the first part describes system architecture, the constraint programming method and the fundamental equations that control the strategies for resources selections. Afterwards an example is presented, to highlight the characteristics of the proposed method.

2. PROBLEM POSITION

The identification of the methods and the processes suitable to machine a product, can be defined as a resources selection problem, which objective is the design of micro-cycles to machine a subset of the part (*Manufacturing Feature* /5/), coupled to the problem of sequencing of the micro-cycles defined. The design of each micro-cycle consists in the cutting parameters and tool-path determination for the single tool, while in the sequencing it is necessary to find priorities among the cycles, respecting the technological constraints and the objective function imposed to the problem.

The resources selection problem is constrained both by Manufacturing Features information (form, dimension, etc.) and by production system capability (productivity, machine precision, etc.); it can be formulated as a CSP. The constraint definition requires an efficient organisation of part and manufacturing system data, as well as an appropriate formulation of the problem. In the proposed approach, an object oriented representation scheme has been used to design part databases (*Features Data-base*) and manufacturing system capabilities (*Resources Database* and *Knowledge Database*) /6,7/. Moreover, a constraint programming approach has been used to transform technological rules, defined in manufacturing handbooks /8/, in constraint expressions with integer variables and Branch-and-Bound solution technique /9/. Figure 1 shows the architecture of the decision making system and its interfaces with the CAD model and the manufacturing system; two types of constraints may be highlighted: the implicit ones are represented by the objects instance in the Resource Databases and defines the resources availability. The explicit ones formalise the knowledge database and are applied to available resources during problem solution; figures 2 and 3 are an example of the knowledge representation that have been used in this paper for hole-making process.

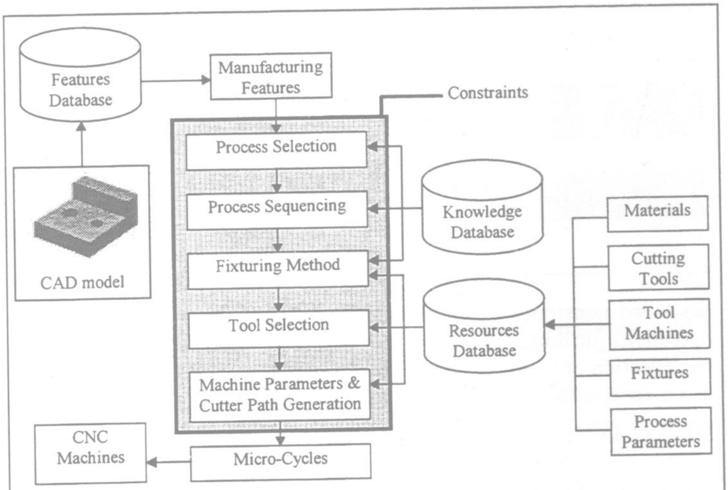


Figure 1

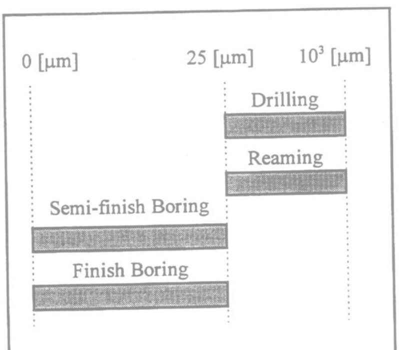
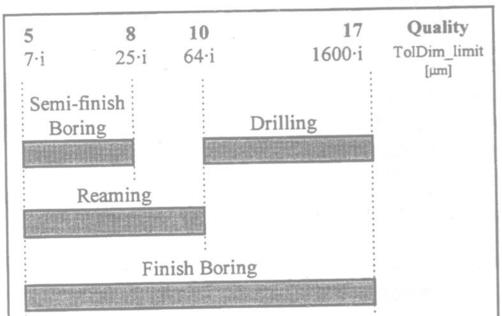


Figure 2



$$i = Tol\_unit = 0.45 \cdot \sqrt[3]{\text{Hole Diameter}} + 10^{-3} \cdot (\text{Hole Diameter})$$

Figure 3

A commercial CAD system (CATIA Version 4.1.7) has been used for the workpiece solid model representation; programs in C++ language have been developed for Object Oriented database design while an high level constraint programming software (ILOG Solver 3.00) has been used to manage constraints.

3. THE RESOURCE SELECTION MODEL

In the approach used each *Manufacturing Feature* **MF_i** is associated to a *Resources Set* **R_i=[MO_i, TM_i, T_i, PP_i, F_i]**, compound of set of *Machining Operation* **MO_i**, *Machine Tools* **TM_i**, *Cutting Tools* **T_i**, *Process Parameters* **PP_i** and *Fixtures* **F_i** /10/ necessary to machine **MF_i**. Constraints are applied to R_i in order to obtain set of resources that respect Manufacturing System Capability and part characteristics (e.g. excluding those machine tools inadequate for the product precision requirements); the result of this phase is a *Constrained Sets* **$(R_i)_{BOUND}$**, sub-sets of R_i which represent problem solutions (Fig. 4). From this sets it is possible to find the *Machining Cycles* **MC_{ij}=[TM_{ij}, T_{ij}, PP_{ij}]** which optimise the objective function defined for the problem (e.g. minimum working time or working cost). In this paper the R_i and $(R_i)_{BOUND}$ definition (selection of machining operations, cutting tools and cutting parameters) has been done for simple holes as Form Features. Following will be described constraint expressions that represent the applications ranges of the possible machining operations (drilling, boring and reaming) in function of position and dimension tolerance information; the selection of tools and cutting parameters is instead conditioned by the raw material characteristics and machines tool power.

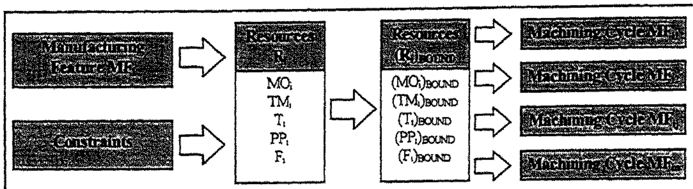


Figure 4

3.1 SELECTION OF THE MACHINING OPERATION

In the hypothesis adopted, the selection of Machining Operation MO_i depends from tolerance and dimensional information assigned to Hole (position and dimension tolerance, diameter and depth). In the case in which the Hole diameter is greater than 50 [mm], milling operation substitutes drilling and these operations have priority respect to reaming and boring. Variability field verification is done using a constraint programming approach with integer variables that can assume values (0,1); an assigned data, for example manu_TruePos (value of Tolerance_Value attribute), belongs to a variability range if the variable value (K_TruePos) linked to this data is equal 1; it is 0 in other case. This procedure is formalised in terms of constraints, using the equation:

$$\sum_{i=0}^{z}(\text{K\_TruePos})_i = 1 \qquad (1)$$

where z is equal 1 because there are two ranges (Fig. 2) and the sum is equal 1, because there is only one range to which the data value belongs. This equation is not still sufficient to solve the variability range problem; the terms $(\text{TruePos\_inf})_i$ and $(\text{TruePos\_sup})_i$ are defined to consider the limits of every range ($(\text{TruePos\_limit})_i$, $(\text{TruePos\_limit})_{i+1}$), with i=0,z:

$$(\text{TruePos\_inf})_i = \left[\text{manu\_TruePos} - (\text{TruePos\_limit})_i\right] \quad (\text{TruePos\_sup})_i = \left[(\text{TruePos\_limit})_{i+1} - \text{manu\_TruePos}\right]$$

These terms are both positive if manu_TruePos belongs to the range. Hence the constraints which complete the problem are:

$$\sum_{i=0}^{z}\left[(\text{K\_TruePos})_i \cdot (\text{TruePos\_inf})_i\right] \geq 0 \qquad (2)$$

$$\sum_{i=0}^{z}\left[(\text{K\_TruePos})_i \cdot (\text{TruePos\_sup})_i\right] > 0 \qquad (3)$$

The same procedure for variability range verification can be applied to Dimension Tolerance value (manu_DimTol), with variables $(\text{K\_DimTol})_i$ and terms $(\text{DimTol\_inf})_i$ and $(\text{DimTol\_sup})_i$ with z=11.

$$\sum_{i=0}^{z}(K\_DimTol)_i = 1 \tag{4}$$

$$\sum_{i=0}^{z}\left[(K\_DimTol)_i \cdot (DimTol\_inf)_i\right] \geq 0 \tag{5}$$

$$\sum_{i=0}^{z}\left[(K\_DimTol)_i \cdot (DimTol\_sup)_i\right] > 0 \tag{6}$$

$(DimTol\_inf)_i$ and $(DimTol\_sup)_i$ are defined in function of $Tol\_unit=0.45\cdot(D^{1/3})+0.001\cdot D$ and $(Tol\_Coeff)_i=\{7,10,16,25,40,100,250,400,640,1000,1600\}$ /11/; the product of these two terms identifies the limits of the Dimension Tolerance ranges, in function of Through Hole diameter (Fig. 2):

$$(DimTol\_inf)_i = \left[manu\_DimTol - Tol\_unit \cdot (Tol\_coeff)_i\right] \quad (DimTol\_sup)_i = \left[Tol\_unit \cdot (Tol\_coeff)_{i+1} - manu\_DimTol\right]$$

If the Hole diameter is less than 50 [mm], drilling operation is always active and it is the only machining operation if manu_TruePos is greater than 25 [μm] and manu_DimTol greater than 64·Tol_unit [μm]. In the other case it is necessary to activate other operations MO_i, in function of the knowledge representation reported in fig 2-3 which is formalised in terms of constraints as:

$$Ream\_Value = (K\_TruePos)_1 \cdot \sum_{i=0}^{4}(K\_DimTol)_i \tag{7}$$

$$Finish\_Bore\_Value = (K\_TruePos)_0 \tag{8}$$

$$Semifinish\_Bore\_Value = (K\_TruePos)_0 \cdot \sum_{i=1}^{2}(K\_DimTol)_i \tag{9}$$

where Ream_Value, Finish_Bore_Value and Semifinish_Bore_Value represent indexes associated to machining operations; if the value is equal 1, the operation becomes active.

3.2 THE SELECTION OF THE CUTTING TOOL AND MACHINING PARAMETERS

The tool selection phase is dedicated to retrieve from *Resources Database* (Fig. 1) the tools sets T_{ij}, linked to operations MO_{ij}, which satisfy geometric and technologic constraints (hole diameter and hole depth, tool machine power). The selection of the drill cut tool diameter considers operative limitations due to presence of other operations (maximum tool diameter must be less than hole diameter when is done before reaming or boring machining), to the existence of a pre-cut hole obtained with other processes, to the tool availability (verified analysing the Drill Cutting Tool class in the Resources Database) and machine tool power. The search algorithm used to identify tool diameter is similar to the one used to verify the tolerances ranges; infact it identifies the range that contains the hole diameter and it chooses the tool diameter equal to lower limit of this range. The problem formulation, using a constraint programming approach, uses the following expressions:

$$\sum_{i=0}^{z}(K\_Diameter)_i = 1 \tag{10}$$

$$\sum_{i=0}^{z}\left[(K\_Diameter)_i \cdot (Diameter\_inf)_i\right] \geq 0 \tag{11}$$

$$\sum_{i=0}^{z}\left[(K\_Diameter)_i \cdot (Diameter\_sup)_i\right] > 0 \tag{12}$$

with z equal to the number of tools in the database minus 1, and the terms $(Diameter\_inf)_i$ and $(Diameter\_sup)_i$ equal to:

$$(Diameter\_inf)_i = \left[Hole\_Diameter - (Diameter\_limit)_i\right] \qquad (Diameter\_sup)_i = \left[(Diameter\_limit)_{i+1} - Hole\_Diameter\right]$$

Identified the maximum tool diameter (**D_{max}**), the following phase is verification of machine tool power in function of the process parameters, as cutting speed v_C, feed **f** and cut pressure **k_C**; these values are retrieved from *Resources Database* in function of the work-piece material, the material hardness and the tool diameter /8/. The constraint programming approach offers the possibility of evaluating different alternative, with advantages of lower times required respect to the IF-THEN approach; for example if the machining power, with tool diameter equal to 16 [mm], is greater than the machine power and the cutting tool diameter set is (2,10,16) [mm], the constraint programming can verify all permutations (2-16, 10-16, 2-10-16) which satisfy the power constraint, but moreover it is able to choose the combination which optimises an objective function. Defining as $\mathbf{P}_{ci}=k_{ci}\cdot(D_i\cdot f_i\cdot v_{ci})/(4\cdot 60)$ /8/ the machining power required using the tool diameter $\mathbf{D}_i$ and $(\mathbf{P}_{cw})_j$ the available power of tool machine TM_j, the adopted optimisation criterion is to find the tool sets which maximise the global efficiency:

$$E = \max\left(\prod_{i=0}^{z}\left[P_{ci}/(P_{cw})_j\right]\right) \tag{13}$$

with z equal to the tool number minus 1. The hypothesis that only two cutting tools are sufficient to satisfy power constraint criterion, can be formulated in terms of constraints, as:

$$\sum_{i=0}^{z}(K\_Power)_i = 1 \tag{14}$$

$$\max\left\{\sum_{i=0}^{z}\left[(K\_Power)_i \cdot P_{ci}\right]\right\}\cdot\left\{\left(\frac{k_c\cdot f\cdot v_c}{4\cdot 60}\right)_{D_{max}}\cdot\left[D_{max}-\sum_{i=0}^{z}\left[(K\_Power)_i\cdot D_i\right]\right]\right\} \tag{15}$$

where **E** is the maximum value of efficiency on the machine TM_j. The same procedure is applied to the tool selection in reaming and boring operations.

4. RESULTS

Fig. 5 shows boundary and drafting representation of a simple part, composed of Form Features (a through step, a through hole and a blind hole) and Precision Features (a true position tolerance and a dimensional tolerance); according to our representation scheme (Fig. 6), they define a single Manufacturing Feature, because the machining of the *Through_Hole* and of the *Blind_Hole* is influenced from the machining of the *Through_Step*, and from the *True_Position* and *Dimension_Tolerance* information. All these information are stored in the Features Database. In the Analysing Phase, the decision making system retrieves information on Manufacturing Feature in order to define Process and Resource Selection identification; these information are the values of Form and Precision features attributes: (*FORM_1+*PREC_1), (*FORM_2+*PREC_2) and (*FORM_3). In function of these data, the decision making system identifies three machining operations, connected to form features of the workpiece: Reaming_Operation to *FORM_1, Boring_Operation to *FORM_2 and Milling_Operation to *FORM_3. Figure 6 shows a part of the representation scheme of classes of CAD/CAPP interface and the flow of information during decision-making process of the Boring_Operation *BORING_1; in this case the system activates two Machining_Cycle: *CYCLE_1 and *CYCLE_2, retrieving the information on the machining from Resources Database and verifying the constraints. *CYCLE_1 is a drilling machining, with cutting tool *DRILL_1 and cutting condition *CUTT_1, while *CYCLE_2 is a boring machining with cutting tool *BORE _1 and cutting condition *CUTT_2.

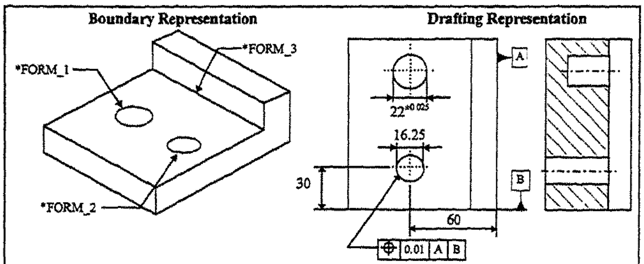


Figure 5

5. CONCLUSIONS

The design and the implementation of a decision-making system for process planning using a constraint programming approach, can efficiently support process planner during resources and processes selection tasks. The most difficult task in this work was the modelling of the technological relations from rules to constraints. The main advantage in using constraints instead of rules is the possibility to implement easily the system adding new constraints on the existing modelling with limited changes. This kind of implementation enables the process planner to transform a general purpose decision making system in a specific one; infact all the information on manufacturing system capability can be referred to the individual company, and product and system limitations can be added in a simple manner.

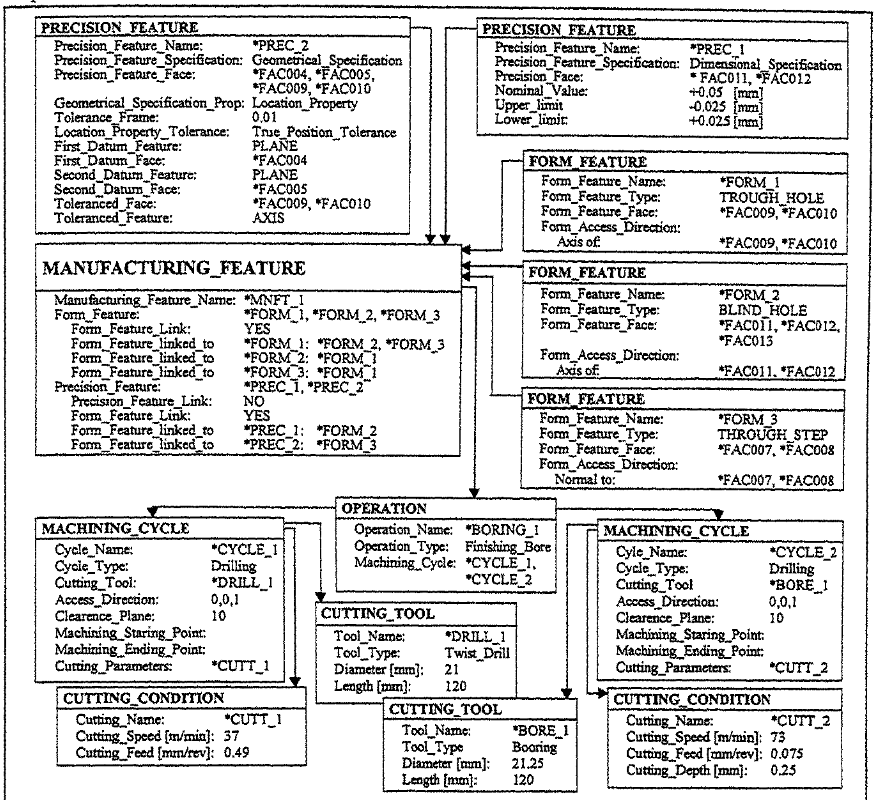


Fig. 6

REFERENCES

1. Algeo, M. B., A State-of-the-Art Survey of Methodologies for Representing Manufacturing Process, NISTIR 5391, National Institute of Standards and Technology, Gaithersburg, MD, 1994.
2. Chang, T.C., Wysk, R.A., and Wang, H.P., Computer-Aided Manufacturing, Prentice-Hall Inc., Englewood Cliffs, NJ, 1991.
3. Parsei ,H. R., Kolli, S. and Hanley, T. R., Manufacturing Decision Support Systems, Chapman & Hall, London, 1997.
4. Puget, J. F., A C++ Implementation of Constraint Logic Programming, ILOG Solver Collected papers, ILOG tech reports, 1994.
5. Tricarico, L., Design of CAD/CAM/CAM interfaces using an Object Oriented Database approach, Research Report, ENS Lurpa, Cachan, F, 1994.
6. De Toma, F., Modellazione di un sistema per la generazione automatica dei cicli di lavorazione con un approccio di tipo object-oriented, Master Thesis, Politecnico di Bari, Bari, IT, 1994.
7. Chep, A., Tricarico, L., Bourdet, P. and Galantucci, L. M.: Design of Object-Oriented Database for the definition of machining operation sequences of 3D workpiece, submitted for publication.
8. Machining Data Handbook, 3rd Edition, Institute of Advanced Manufacturing Sciences INC., Cincinnati, 1980, Vol. 1-2.
9. ILOG Reference Manual, Version 3.0, ILOG, 1995.
10. Wysk, R. A., Peters, B. A. and Smith, J. S., A formal process planning schema for shop floor control, Engineering Design and Automation, Vol. 1, N° 1, pp. 3-20, Spring 1995.
11. UNI Norm 6387-68

AUTOMATIC FEATURE RECOGNITION FROM SOLID MODELS

T.N. Wong and S.M. Lam
Department of Industrial and Manufacturing Systems Engineering
The University of Hong Kong
Pokfulam Road, Hong Kong

ABSTRACT
This paper presents a new algorithm for machining feature recognition from the Boundary Representation (B-Rep) solid modeller. The algorithm is based on the construction of a graph to denote a sequence of edges and faces of the solid object. Concave edges are the main attribute of the graph which are arranged in a sequence according to the Next-Next-Edge (NNE) adjacency relationship, which is in accordance with the half-edge data structure of the B-Rep modeller. Faces adjacent to the edges of the sequence are then attached to the sequence to construct the complete graph, named Face-Edge-Sequence (FES) in our algorithm. It is found that the FES graphs for some primitive features, such as slot, step, through hole, etc., can easily be interpreted and recognized. The proposed method is also efficient in dealing with some cases of feature interactions.

KEYWORDS
feature recognition, face-edge-sequence, boundary representation, half edge data structure

1. INTRODUCTION

Feature recognition has been regarded as an important approach to support integrated manufacturing. The purpose of feature recognition is to recognize and extract high level features (such as machining features of a part) from a geometrical model, which provides low level geometric and topological information (vertex, line, face) from the design stage. With respect to machining, the output from feature recognition is a set of machined features such as slot, key, or blind hole which are "informationally complete" from the process planning function. Much research efforts have been spent on recognizing features from either CSG or B-rep based solid models, and B-rep model have been considered to be more favourable for feature recognition [1]. However, the majority of contemporary feature recognition systems can handle only a limited number of features. The recognition and interpretation of interacting features are, in particular, still one of the major problems in automatic feature recognition. This paper presents a new graph-based feature recognition method which is capable of recognizing machining features from prismatic parts. Geometric and topological information are based on boundary representation of the features. The method is based on the construction of a graph to denote a sequence of edges and faces of the solid object. The graph, which embraces sets of concave edge nodes and adjacent faces nodes, is called the Face-Edge-Sequence (FES) graph. It is found that the FES graphs for some primitive features, as well as interacting features, can be effectively identified.

2. REVIEW OF RELEVANT WORK

Different approaches have been proposed to tackle the problem of machining feature recognition. Major recognition techniques for B-rep models can be classified into syntactic-pattern methods, rule-based methods, graph-based methods, and volume-decomposition methods. Comprehensive reviews can be obtained in the literature [2-5], only some typical examples are mentioned in this paper. In graph-based approach, Joshi and Chang[6] developed the Attributed Adjacency Graph (AAG) which makes use of the face and edge adjacency of the feature to achieve the recognition purpose. De Floriani [7] proposed the Generalised Edge-Face Graph (GEFG) which provides a face-based topological description of the boundary. Gavankar and Henderson [8] proposed an approach which involved identification of faces with multiple edge loops as candidates for entrance faces of features. Tseng and Joshi [9] proposed a method in recognizing interacting features with multiple interpretations. Ferreira and Hinduja [10] proposed a technique in dealing with 2.5D features. The technique considers the edges of the convex hull of the component's faces and also the inner loops of the edges. In decomposition approach, Sakurai [11] proposed another volume decomposition method for the recognition of polyhedral features, the method decomposes a polyhedron into maximal cells by intersecting it with half spaces of its faces having concave edges. Woo [12] proposed the Alternating Sum of Volumes (ASV) which uses a convex hull approach to generate difference volume from a convergence feature. Kim [13] proposed another convergent convex decomposition called Alternating Sum of Volumes with Partitioning (ASVP) that combines the ASV decomposition and remedial partitioning for the non-convergence features. Mainly due to the reason of non-uniqueness in solid representation, feature recognition is not easy for CSG based models [1]. Among the few reported feature recognition techniques for CSG models, Lee and Fu [14] proposed an approach based on principal axis and tree reconstruction to extract and unify manufacturing features. Perng et al [15] proposed an algorithm that converts a CSG tree representation into a DSG (Destructive Solid Geometry) tree representation. The DSG is a specific form of CSG which all regularized set operators are of difference operation that it is a machining feature oriented representation.

3. FACE-EDGE SEQUENCE

The construction of FES is based on the half-edge data structure of boundary representation of solids. The half-edge data structure was first proposed by Mantyla [16] in his GWB modeller. It is a variation of the full winged-edge data structure originally proposed by Baumgart [17]. The most significant drawback of winged-edge data structure, as stated by Kalay [18], is its dual rule in bounding the faces and connecting adjacent faces that makes the edge traversal complicated. The half-edge data structure solves such problem by breaking the edge into two halves as shown in Fig. 1. Each edge is represented by three records : two half-edge records and one edge record. Each edge record has only one adjacent face, and their connectivity is maintained by adding an explicit pointer in each half-edge record that references the edge record. The edge record, on the other hand, consists of the left and right half-edge. Consequently, the edge associates two half-edges with each other.

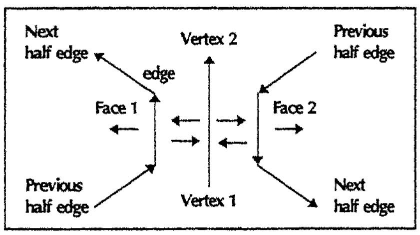


Fig. 1 : Half-Edge Data Structure

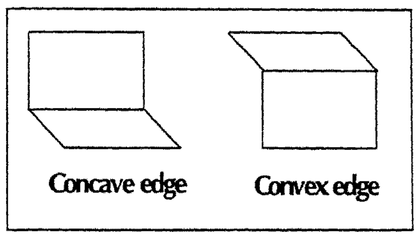


Fig. 2 : Concave and Convex Edges

Concave edges, convex edges and faces are the attributes for the construction of FES sequence (Fig. 2). The sequence makes use of the topological information between edges and faces to define features. The sequence has two kinds of nodes : one is edge node and the other is face node. Each sequence corresponds to edges in the same direction (same principle axis direction). Taking the example of a simple slot (Fig. 3). There are two concave edges [E2, E3], two convex edges [E1, E4], and faces [F2, F2, F3]. To construct the FES, it first points to one of the concave edges arbitrarily, say E2. It then searches for the next-next edge (NNE) from E2 and it selects E1 (as both E1 and E2 belongs to face F1). As E1 is a convex edge, it becomes the starting node of the sequence. Then the pointer searches for the NNE from E2 again on the next face F2 and it selects E3 which is concave. The pointer will then point to the NNE for E3 on the next face F3, and E4 is found and it becomes the ending node of the sequence since E4 is a convex edge. A sequence of [E1 - E2 - E3 - E4] is formed which corresponds to nodes [V - C - C - V] (V = convex edge; C = concave edge). Next, it has then to search for the faces which are topologically connected to the edge nodes. The topological relationship is found from the boundary representation of the solid. In this way, face node [F1] is found to be connected to the edge nodes [E1, E2], face node [F2] is connected to the edge nodes [E2, E3], face node [F3] is connected to edge nodes [E3, E4]. Similarly, a simple step also has a FES [V - C - V].

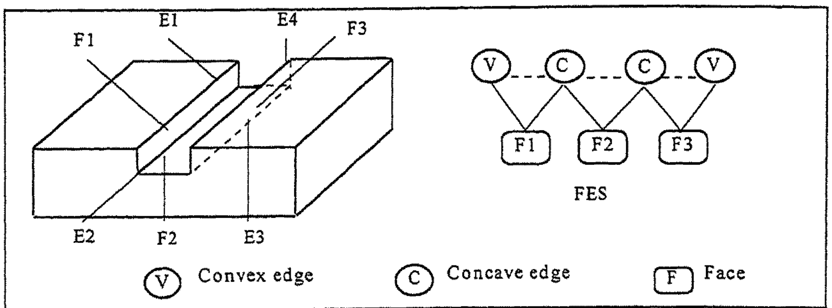


Fig. 3 FES graph for a simple slot

In dealing with features of end-step, key and blind hole, a network of FES is always formed. These features always have concave edges in three directions and therefore, three independent FES are formed. Figure 4 shows some examples of features with three direction of edge sequences. For the key, concave edges include [E1, E2, E4, E5, E11]; convex edges [E3, E6, E7, E8, E9, E10]; faces [F1, F2, F3, F4,]. Concave edges [E4] belongs to the x-axis direction, [E2, E11] belong to y-axis direction and [E1, E5] belong to z-axis direction. They form 3 independent FES : [E3 - E4 - E9] corresponds to [V - C - V]; [E1 - E2 - E11 - E8] corresponds to [V - C - C - V]; [E6 - E7 - E5 - E10] corresponds to [V - C - C - V]. By obtaining the faces topological information, a network of FES graph for the key is

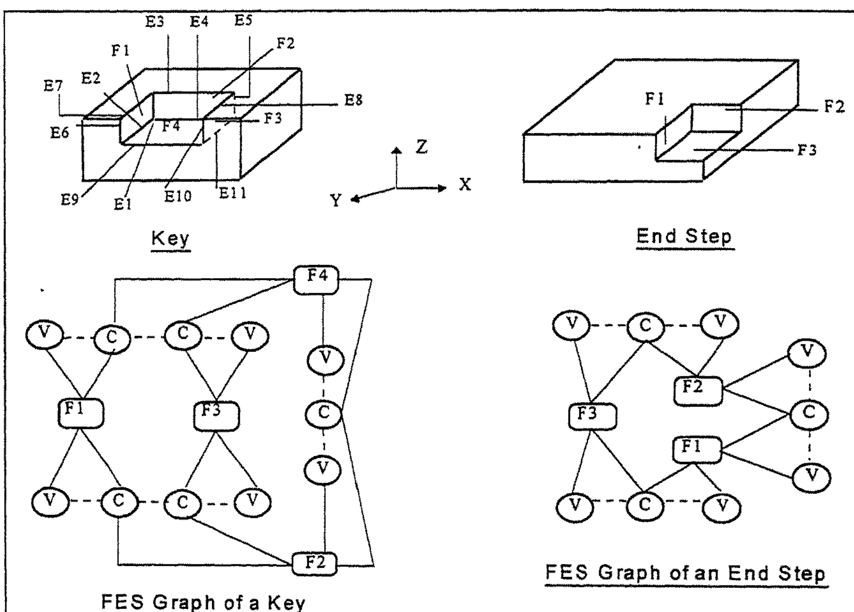


Fig. 4 : Features with FES in three directions

formed. The current implementation of FES algorithm can recognise 6 types of primitive features : step, slot, key, end step, through hole and blind hole. Step, slot and through hole have one direction of sequence while the others have three directions of concave edges. Multiple features with interaction are recognised as composition of these 6 primitive features.

4. MULTIPLE FEATURES

Using the FES algorithm, feature identification is based on the edge information. Features that do not interact at the concave edges can be easily identified. The FES algorithm deals with features with inner loop efficiently. The algorithm will isolate the inner loop feature from the stock to carry out the recognition process. In Fig. 5(a), the pocket will first be isolated as an inner loop feature. The step is independent from the pocket as there is no interaction between the edges of the step and pocket. Independent FES graphs for pocket and step are therefore found. As depicted in Fig. 5(b), although the three concave edges are in the same direction, they cannot join to form a single sequence. Independent slot and step are recognised.

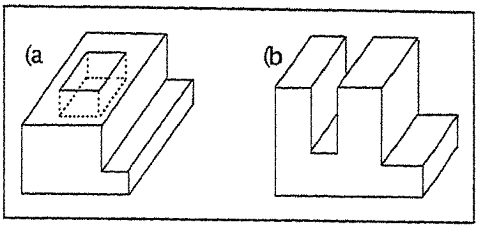


Fig. 5 : Multiple Features

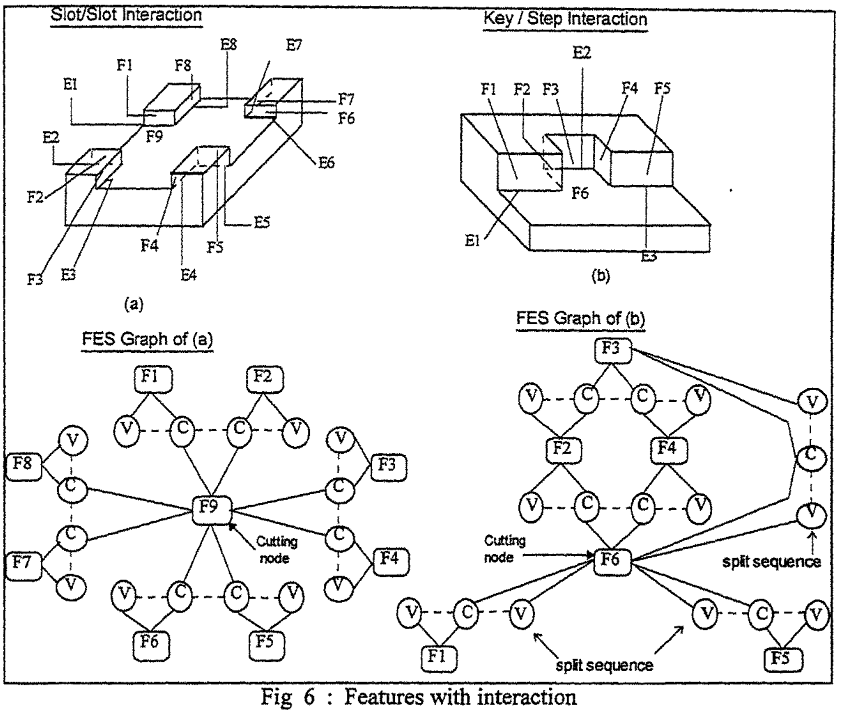


Fig 6 : Features with interaction

The identification of features with interaction is a main concern in the recognition process. Complex features with interaction are usually formed by the intersection of some primitive features. Some of the geometric information will lose in the interacting features. The FES algorithm is capable of reducing a multiple and complex interacting feature into its constituting primitive features. For the cases of feature interaction illustrated in Figure 6, the interacting features share a common face. Fig. 6(a) depicts the case of slot-slot intersection. Concave edges [E1, E2, E5, E6] and [E3, E4, E7, E8] are in two edge groups according to their directions. They all share the common face F9. Consequently, four sequences are formed : [V - E1 - E2 - V]; [V - E5 - E6 - V]; [V - E3 - E4 - V]; [V - E7 - E8 -V]. From the FES graph, [F9] is the face node that is shared by the four sequences. In this case, the face node [F9] becomes the cutting node of the graph as all four independent sequences are connected to this node. Therefore, the interacting features in Fig. 6(a) are recognised as four slots that formed by faces : [F1, F2, F9]; [F3, F4, F9]; [F5, F6, F9]; [F7, F8, F9]. Another example of recognising features with interaction is shown in Fig. 6(b), showing the concave edges and adjacent faces. In this case, the features have concave edges in three directions. Concave edges [E1, E2, E3] are in the same direction and are all adjacent to the common shared face F6. A sequence [E1 - E2 - E3] is found such that E2 is the NNE of E1 and E3 is the NNE of E2. However, such sequence is not valid in the construction of the FES graph. It is an open sequence, and therefore, F6 is an open shared common face. A split process is then carried out to isolate the edge nodes from the sequence. Each isolated edge will search its NNE from its another adjacent face, and the edge node is still connected to the shared common face by adding a virtual convex edge node. Then, three independent sequences are formed. In the final FES graph, face node [F6] is the cutting node that 1 key graph and 2 step graphs are formed. Therefore, the features are recognised as 1 key with composing faces [F2, F3, F4, F6] and 2 steps with composing faces [F1, F6] and [F5, F6].

In some cases of interacting features, the shared common face may be found both "open" and "close". In Fig. 7, the feature has concave edges in three directions. Concave edges [E2, E3, E5, E7, E9, E10] are in the same direction (direction y) that they share the common face F3. These concave edges form an close loop. Therefore, F3 is termed "close" in this direction (direction y). Edge nodes forming a close loop have to be deleted from the final FES graph, the faces adjacent to these edges are also recorded. Another direction of concave edges [E1, E4, E6, E8, E11] share the common face F3 (direction x). These edges form an open loop and therefore, F3 is termed "open" in this direction. A split process is carried out to divide the chain [E1, E4, E6, E8, E11] into five individual sequence [V - C - V]. In the construction of FES graph, only concave edges of direction x and z are used as the sequence in the y-direction is a closed loop. From the final FES graph, 1 blind hole (composing edges [E4, E5, E6, E8, E9]) and 2 end steps (composing edges [E1, E2]; [E10, E11]) are recognised. It is found that the FES graphs for the primitive features recognised are the sub-graphs of the original FES graphs because one direction of concave edges have been deleted. In some cases, 2 groups of concave edges, in different directions, may be deleted from the graph. From the above feature, a slot [F1, F2, F3] is recognized which does not appear in the final FES graph. In the construction of the final FES graph, one direction of concave edges (y direction) is deleted. The faces adjacent to the concave edges in this deleted sequence is recorded and examined whether such faces appear in the final FES graph. As [F1, F2] do not appear in the final FES graph, they are checked for the slot formation. The conditions for slot formation are :

- the faces are connected to the shared common face with concave edges;
- their face normals are parallel and opposite in direction.

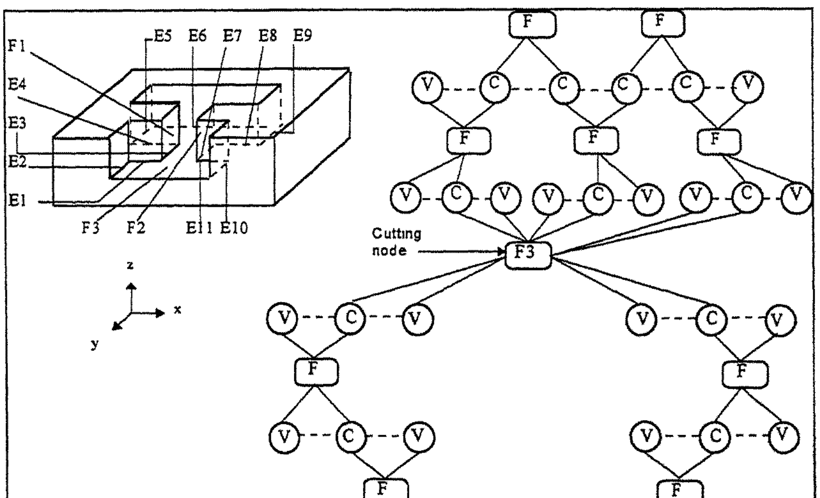


Fig : 7 Multiple features with 1 blind hole, 2 end steps, and 1 slot

5. CONCLUSION

Face-Edge-Sequence (FES) provides an efficient method in recognising features. The geometrical and topological information of solid objects are adequate based on the B-Rep solid modeller. Basically, a FES graph is formed by searching for concave edges and adjacent faces of the features. The algorithm avoids exhaustive search of edges and faces that are not directly adjacent to the features required. FES graph can efficiently interpret and recognise six primitive features. Based on the edge information, features that have no interaction with concave edges can be easily identified. The identification of features with interaction is a main concern in the feature recognition process. The proposed method is also efficient in dealing with feature interaction. Using the proposed method, the shared common face of interacting features is the face node with which all different feature graphs are joined. It is then possible to isolate the different feature graphs from this face node, and hence the multiple features are recognized as a number of primitive features. Current implementation of the FES algorithm can deal with polyhedral features, the work is being extended to the recognition of more complex interacting cases.

6. REFERENCE

1. Woodward, J.R. "Some speculations on feature recognition", Computer Aided Design, Vol. 20, No. 48, pp. 189-196, 1988.
2. Shah, J.J., "Current status of features technology", CAM-I report R-88-GM-04.1, 1988.
3. Shah, J.J., "Assessment of features technology", Computer Aided Design, Vol. 23, No. 5, pp. 331-343, 1991.
4. Bronsvoort, W.F. & Jansen, F.W., "Feature modelling and conversion- Key concepts to concurrent engineering", Computers in Industry, Vol. 21, pp. 61-86, 1993.
5. Wu, M.C. & Liu, C.R., "Analysis on machined feature recognition techniques based on B-rep", Computer Aided Design, Vol. 28, No. 8, pp.603-616, 1996.
6. Joshi,. S., and Chang, T. C., "Graph-based heuristics for recognition of machined features from a 3D solid model", *Computer Aided Design*, Vol. 20, No. 2, pp.58-65, 1988.
7. De Floriani, L,, "Feature extraction from boundary models of three-dimensional objects", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 11, No. 8, pp.785-798, 1989.
8. Gavankar, P., and Henderson, M. R., "Graph-based extraction of protrusions and depressions from boundary representations", *Computer Aided Design*, Vol. 22, No. 7, pp.442-450, September 1990.
9. Joshi, S., and Tseng, Y. J., "Recognizing multiple interpretations of interacting machining features", *Computer Aided Design*, Vol. 26, No. 9, September 1994.
10. Ferreira, J. C. E., and Hinduja, S., "Convex hull-based feature-recognition method for 2.5D components", *Computer Aided Design*, Vol. 22, No. 1, pp.41-49, Jan/Feb 1990.
11. Sakurai, H., "Volume decomposition and feature recognition : Part 1 - polyhedral objects", *Computer Aided Design*, Vol. 27, No. 11, pp.833-842, 1995.
12. Woo, T. C., "Feature extraction by volume decomposition", *Proceedings of Conference of CAD/CAM in Mechanical Engineering*, Cambridge, MA, USA, pp. 39-45, 1982.
13. Kim, Y.S., "Convex decomposition and solid geometric modeling", Ph.D. Dissertation, Department of Mechanical Engineering, Stanford University, 1990.
14. Lee, Y.C., Fu, K.S., "Machine understanding of CSG: Extraction and unification of manufacturing features", IEEE Computer Graphics and Applications, pp. 20-32, January 1987.
15. Perng, D.B., Chen, Z., and Li, R.K., "Automatic 3D machining feature extraction from 3D CSG solid input", *Computer Aided Design*, Vol.22, No.5, 1990.
16. Mantyla, M.,"An introduction to solid modelling", *Computer Science Press*, Maryland, USA, 1988.
17. Baumgart, B. G., "Geometric modelling for computer vision", Ph.D. thesis, Department of Computer Science, Stanford University, 1974.
18. Kalay, Y.E., "The hybrid edge : a topological data structure for vertically integrated geometric modelling", *Computer Aided Design*, Vol. 21, No. 3, pp.130-140, April 1989.

MULTI-VIEW FEATURE MODELLING FOR MANUFACTURING APPLICATIONS

T.N.Wong & C.B. Leung
Department of Industrial & Manufacturing Systems Engineering
The University of Hong Kong

ABSTRACT

The mapping of design intent to the subsequent downstream application features, such as machining planning, setup planning, engineering analysis, assembly planning, etc., is an important area of research in CIM. Feature technology is recognized to have the potential for automating and integrating design and the downstream applications in the CIM environment. Much research efforts in features have been contributed in the area of linking design and a downstream application such as manufacturing process planning using either the feature recognition or the design by features approaches. However, only a limited number of feature based systems can link up more than one applications. This paper describes a prototype feature conversion system for integrating feature based design with multiple downstream applications. In the prototype system, there are two types of features : neutral features and application features. The system is armed with algorithms to convert between neutral features and application features.

KEYWORDS

Feature Conversion, neutral features, machining, setup planning

1. INTRODUCTION

In the last decade, a lot of research efforts in CAD/CAM have been spent on the development of feature technology, endeavoring to automate and integrate the design, analysis and manufacturing functions. Majority of the studies in features have been concentrated on feature recognition and design by features. Contrast to these two approaches, feature conversion is a relatively new approach of feature technology. Different terms have been used to identify the feature conversion process, for example, feature refinement, feature conversion, feature mapping, and feature transformation [1][2][3]. Features have multiple views. For the same workpiece, different applications have different feature models involving different views, definitions and explanations of features. Hence, each application has its own set of features for the same workpiece. Based on this concept, a user can design the workpiece by using design features. Then, different features for different applications can be converted from the set of design features.

Most of the feature-based systems are developed for a particular application and only one feature model is associated with the geometric model. Hence, a feature representation scheme is designed to suit a downstream application such as process planning. For example, different feature representation schemes and classification systems have been proposed by researchers including Gindy et al [4][5], Ssemakula and Satsangi [6], Sheu and Lin [7], Kang and Nnaji [8], Hummel [9] and Gu [10]. All these classification schemes are face based, and they are basically used in feature based systems for integrating the design and process planning functions. However, it is difficult to adopt this kind of feature taxonomies in feature conversion systems as a feature conversion system has to support more than one applications.

In contrast to feature recognition and design by features, very little work has been done on feature conversion. Typical examples of feature conversion systems are the approaches proposed by Shah [11], Fu et al [12] and Bronsvoort and Jansen [3]. Defining the relationships between feature spaces, Shah used the concept of feature transformation to develop a system for mapping between conjugate spaces [10]. This system provides generic functions for data extraction and geometric reasoning. A feature space is a combination of application, product type and level of abstract. It can be thought as n-dimensional vector spaces where n is the maximum number of independent vectors. Number of dimensions increases as the complication of the space. For instance, the number of dimensions of a form feature space is larger than that of a wireframe geometry space. The relationship of different spaces were identified. They are overlapping spaces, projected spaces, conjugate spaces, adjoint spaces. In addition, transformations between spaces were identified, including the identity transformation, adjoint transformation, projection transformation and conjugate transformation. Overlapping of spaces was considered in transformations. If two sub-spaces are fully disjoint, transformation is not available.

Fu et al [12] proposed a feature representation scheme which is based on graph grammar. Inside the graph structure, a node represents a geometric primitive (such as a solid primitive or a surface) and the links represent the constraint among primitives such as distance or angle. Graph grammar parsing is combined with knowledge-based techniques to derive feature information and propagate constraints. The proposed approach can be used for the transformation of features. It involves the process of rewriting a graph by embedding the transformation designated in graph rewriting productions. In addition to the proposed face-based representation, two types of additional information are provided. They are parametric object representation and a CSG tree.

On the other hand, Bronsvoort and Jansen [3] identified four types of feature transformations, namely, identity transformation, projection transformation, adjoint transformation and conjugate transformation. Their work involved in extending the GeoNode feature modeling system to incorporate facilities for multiple feature views and feature conversions between arbitrary views. The distinguished point of their system is the use of arbitrary view rather than the primary design view and the secondary application views. A geometric tree with coordinate systems and a procedural representation is used to determine the features in each view.

This paper describes a prototype feature conversion system for integrating feature-based design with the machining planning and the setup planning functions. There is a specific feature model for each application, in addition, a neutral feature model is introduced. Feature conversion among various applications is achieved through a two way conversion process.

2. NEUTRAL FEATURE

In this system, there are two categories of features, namely, neutral features and application features. As the system is currently supporting design, machining and setup planning applications, there are three types of application features, that is, feature-based design features, machining features and setup planning features, corresponding to each application respectively. Conversely, a neutral feature is independent of any applications, it is defined as a three dimensional object bounded by one or more surfaces with specific geometric information including the location and orientation, as well as topological entities such as surfaces and edges. There is no limit on the number of surfaces that bounded a neutral feature. The shape is limited to any possible 3D volume closed by any combination of these type of surfaces. All the application features can be represented by a combination of a subset of the set of all neutral features. It is designed that there exists conversion from the model of neutral feature to all the application feature models and vice versa.

2.1 Classification of Neutral Features

Object oriented concept was applied in designing the system structure. A number of subclasses are defined to store the information for neutral features. The subclass inherit the properties of its parent class(es). The object oriented hierarchy of neutral features are shown in Figure 1. There are 2 ways to classify the neutral features. The first way is classification by shape. The neutral features are divided into 2 main classes: orthogonal boxes and non-orthogonal boxes. The algorithms and design of these 2 classes of neutral features are very different. As implied from the name, all the orthogonal box features are in box shaped. Planar surfaces are used to construct the orthogonal box. Orthogonal boxes are bounded by 6 planar surfaces. Their orientations are orthogonal relative to the world coordinate. This particular class is established because it is a special case that the calculations are very fast and simple. On the other hand, the design of the non-orthogonal box neutral features are based on the 5 types of surfaces.

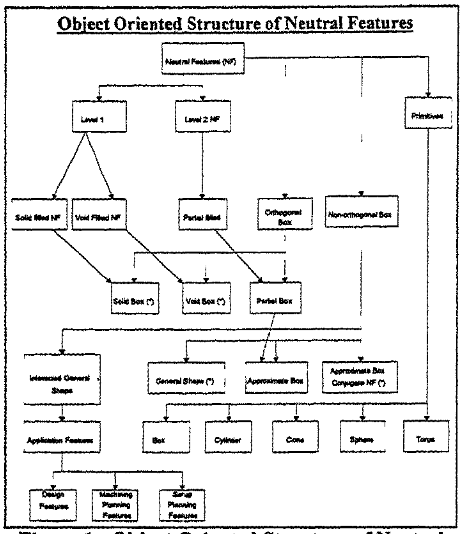


Figure 1 : Object Oriented Structure of Neutral Features

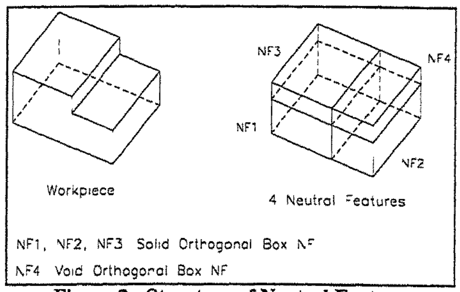


Figure 2 : Structure of Neutral Feature

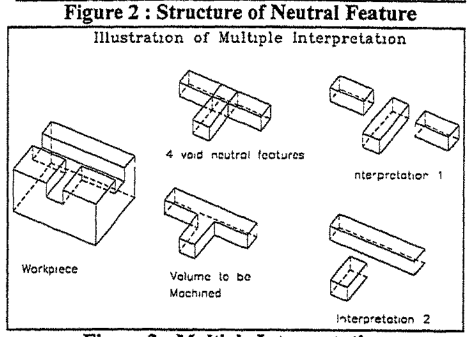


Figure 3 : Multiple Interpretations

The second way is based on the object type and the neutral features can be categorized into either solid or void. The workpiece is represented by all the solid neutral features. That is, the solid representation of the workpiece is equal to the union of all solid neutral features. On the other hand, the union of all void neutral features will result in the solid representation of subtracting the workpiece from its orthogonal envelope.

2.2 Structure of Neutral Features

According to the object oriented description, neutral features are with attributes to describe the properties of the feature. The following code shows the structure of an abstract neutral feature class.

```
(defclass NEUTRAL_FEATURE (is-a FEATURE)
      (slot trans3d [4 X 4 transformation matrix])
      (slot param [list of parameters])
      (slot shape [shape])
      (layer [AutoCAD layer])
      (sol_id [solid ID integer])
      (faces [list of faces])
      (faces_type [list of types of the above faces])
      (face_n_value [list of neighbour value of the above faces])
            ; value ∈ {SOLID, VOID, NIL, PARTIAL}
      (ingredient_nf_list [list of neutral features])
)
```

Slot 'trans3d' stores the location and orientation relative to world in matrix form. The 'param' keeps the parameter of the neutral features. For instance, if this neutral feature is a simple cylinder, the parameter is a list of 2 values, the radius and the height. Regarding the slot 'shape', available values box, cylinder, cone, sphere, torus and general. For better organization, different objects are stored in different layers. For each of these objects, an solid is created and the corresponding solid ID is shown in slot 'sol_id'. The topological information of this solid is kept in the slots 'faces', 'faces_type' and 'face_n_value'. Slot 'ingredient_nf_list' stores the objects that construct this neutral feature. In other words, the union of the objects in this list is the solid representation of the neutral feature. A simple illustration of neutral feature is shown in Figure 2. The workpiece is a box shaped solid with a smaller box removed. There are four neutral features, named NF1, NF2, NF3 and NF4. The first three are solid neutral features while NF4 is void. The union of the first three results in the solid representation of the workpiece. Multiple interpretations of the feature are possible (Figure 3).

3. APPLICATION FEATURES

Similar to neutral feature, application features are divided into 2 classes which are orthogonal box shape and non-orthogonal box application features. The orthogonal box application features are constructed by solid or void orthogonal boxes. Design features are classified into 3 groups : starting features, positive features and negative features (Figure 4). To start a new design, the designer has to construct a starting feature first and then instance the appropriate positive or negative design features onto the stock to generate the geometry of the product. Machining planning features (Figure 5) are the volumes to be removed from stock by a machining process and is therefore a negative feature.

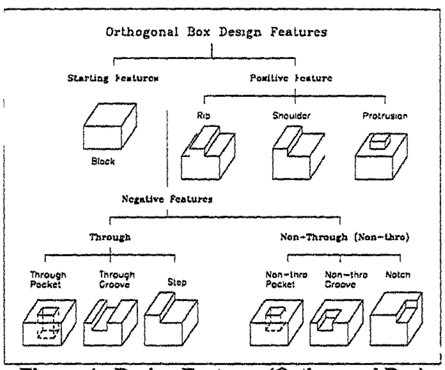


Figure 4 : Design Features (Orthogonal Box)

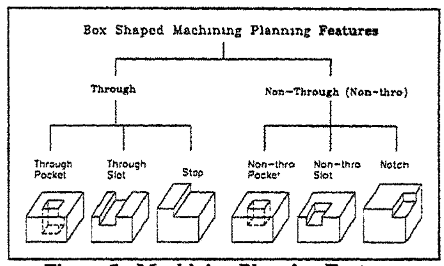


Figure 5 : Machining Planning Features (Orthogonal Box)

A basic setup planning feature is a surface available for location and clamping and is coined as an 'available surface' (AS) in our system. Obviously, setup planning features are surface-based features which are different from all the other volume-based feature types defined in this system such as neutral features and machining planning features. As the feature conversion process deals with volume-based features, a higher level setup planning feature class, 'available surface volume' (ASV), is defined. An ASV is a volume-based feature which is the union of a set of neutral features. At least one of surfaces of an ASV must be open and accessible, that is, the surface is in a position that its outward surface normal points to the empty space and not to any of the other neutral features. Such a surface is actually an AS which can be accessed for setting-up purposes. An illustration of ASV and AS is given in Figure 6.

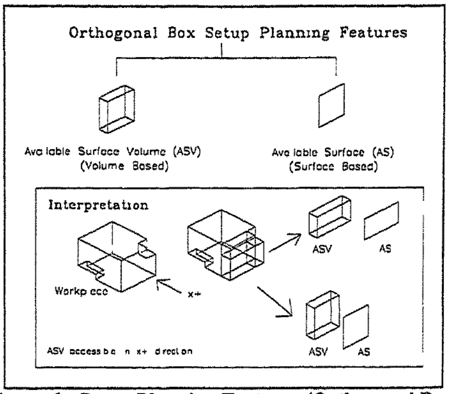


Figure 6 : Setup Planning Features (Orthogonal Box)

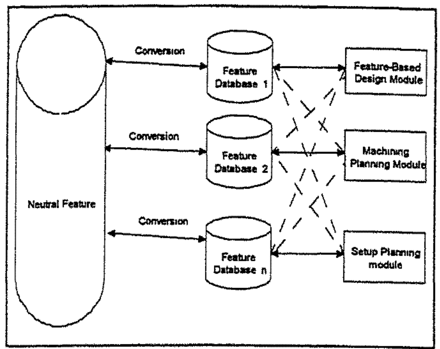


Figure 7 : Architecture of the system

4. FEATURE CONVERSION

Application dependent attributes are provided by the application features. Each application has its own feature domain and feature database. Whenever there are changes in a particular application feature database, corresponding changes in the neutral feature database will be effected. Subsequently, changes in the neutral feature database will trigger changes in application features for all applications. Therefore, the feature conversion process actually comprises several sub-processes, which are, addition of an application feature, conversion from application feature to neutral features, rebuilding of neutral features, verification of all existing application features and the subsequently conversion from neutral features to application features.

Based on the object-oriented technique, each feature is an object with generic data attributes and object-oriented messages for feature manipulation. As all the application features defined in this system are volume-based, whenever an application feature is added to the workpiece and a set of neutral features are converted from this application feature, the overall three dimensional shape of the workpiece is then modified, so as the geometrical and topological information. Since the workpiece is represented by the neutral features, this change in shape will result in the creation of new neutral features, deletion and modification of existing neutral features. Therefore the neutral feature database has then to be rebuilt to incorporate the changes. After this, it is then necessary to verify all the application features in the databases to maintain the consistence of feature representations. It is because some application features may no longer valid due to the deletion or modification of some neutral features during the rebuilding process. Lastly, new application features can be converted from the neutral features database.

5. SYSTEM IMPLEMENTATION

In the development of the prototype system, the AME solid modeler of AutoCAD 12 is used as the geometric modeler and an expert system programming tool Clips 6.0 is used for the development of the knowledge-based system. Combining object oriented programming and expert system technologies; rules are defined to search through some particular classes rather than the facts list of the expert system. This approach is much better than a pure expert system. First of all, the information of the workpiece can be kept in a central object oriented database. This database can be accessed by using both the object oriented techniques, that is messages, and expert system rules-based matching. The structure of the prototype system is presented in Figure 7.

6. CONCLUDING REMARKS

The use of neutral features allows the conversion of features among the design process and different downstream applications. Features of all applications can be converted from the neutral features. For each application, the application features will be stored in its own feature database for further reasoning or processing. When an application modifies its own features, the corresponding neutral features will be adjusted. This modification will be propagated to all the other applications through the conversion process. In other words, feature conversion among different applications can be achieved through this indirect conversion process: modification of a particular application will modify the corresponding neutral features, which will then trigger the modifications of the other application features. This enables the integration of different downstream applications in the CIM environment.

Feature conversion in the system is effected with the rule-based expert system approach, that is, searching through the neutral feature database to look for any pattern which forms the features for the downstream applications. Currently, the machining planning and setup planning applications are implemented in the conversion system together with a feature-based design function.

7. REFERENCES

[1] J.J. Shah, et al, Current status of features technology, CAM-I report R-88-GM-04.1 (1988)

[2] D.C. Anderson and T.C. Chang. Geometric Reasoning in Feature-based Design Planning. *Computers and Graphics* **14,** No.2, 225-235 (1990)

[3] W. F. Bronsvoort and F. W. Jansen. Feature Modeling and Conversion -- Key Concepts to Concurrent Engineering. Computers in Industry. 21, 61-86 (1993)

[4] N.N.Z. Gindy, X.Huang and T.M. Ratchev. Feature-based Component model for Computer-Aided Process Planning Systems. *International Journal of Computer Integrated Manufacturing,* **6,** No. 1 & 2, pp.20-26 (1993)

[5] N.N.Z. Gindy. A Hierarchical Structure for Form Features. *International Journal of Production Research.,* **27,** No. 12, 2089-2103. (1989)

[6] M. E.Ssemakula and A. Satsangi. Application of PDES to CAD/CAPP Integration. *Computers and Industrial Engineering,* **18,** No. 4, 435-444 (1990)

[7] L-C Sheu and J. T. Lin. Representation Scheme for Defining and Operating Form Features. *Computer-Aided Design.* **25,** No.6, June, .333-347 (1993)

[8] T-S Kang and B. O. Nnaji. Feature Representation and Classification for Automatic Process Planning Systems. *Journal of Manufacturing Systems.* **12,** No.2, 133-145

[9] K.E. Hummel. Coupling Rule-Based and Object-Oriented Programming for the Classification of machined Features. *Computers in Engineering ASME 89.* 409-418

[10] P. Gu. A Feature Representation Scheme for Supporting Integrated manufacturing. *Computers and Industrial Engineering,* **26,** No.1, 55-71 (1994)

[11] J. J. Shah. Feature Transformations between Application-Specific Feature Spaces. *Computer-Aided Engineering Journal.* **5,** No.6, December, 247-255 (1988)

[12] Z. Fu, A. De Pennington and A. Saia. A Graph Grammar Approach to Feature Representation and Transformation. *International Journal of Computer Integrated Manufacturing.* **6,** No. 1 & 2, 137-151 (1993)

BACKWARD GROWING-BASED GEOMETRIC REASONING FOR MANUFACTURING FEATURE RECOGNITION

Zhi-Xin Yang[a*], *Ajay Joneja*[b], *Ji Zhou*[c]

[a,*] ieyzx@usthk.ust.hk (Corresponding author), [b] joneja@usthk ust.hk,
Dept. of IEEM, HKUST, Clear Water Bay, Kowloon, Hong Kong.
[c] School of Mechanical Science & Engg, HUST, Wuhan, 430074, P. R. China

ABSTRACT

The paper describes a new methodology for recognition of machining features from a CAD model of complex prismatic parts. The method is fast, and efficiently generates features which have specific machining implications for 3-axis machining centers. The system can handle feature interactions via a set of geometric methods; it generates meaningful part interpretations in terms of feature primitives. The best interpretation is selected by downstream setup planning functions to determine the manufacturing plan for the part. An integrated CAD/CAM system, OSCAP, is developed, based on these ideas.

KEYWORDS: *feature recognition, CAPP, CAD, CAM, backward growing*

1. INTRODUCTION

Feature recognition, which interprets CAD information in a manner that makes sense for CAM, is a weak link in most implementations of integrated CAD/CAM systems for machining. Many researchers [Woo 82, Kim 92, Henderson 84, Gavankar 95, Joshi 88, Wang 93, Karinthi 91, Gupta 94, Vandenbrande 93] have searched for geometric algorithms which will help in this transformation of the design specifications of a part into a set of manufacturing features. Broadly, the following issues have not been adequately addressed by any single approach:

(1) The problem of determining which shape of a part to grow back is non trivial. Not only the optimal stock shape, but the initial orientation of the part could also play an important role in the stock size and the material removal volumes.

(2) To generate interpretations of parts as collections of features, all the recognized features must correspond directly to manufacturing operations.

(3) A clear and efficient searching path for the extraction. This is important not only computationally, but must also be guided by the designers' intent, especially in recognizing features whose topology has been corrupted due to intersections with other features.

(4) Most existing systems cannot recognize feature instances where there are arbitrary intersections between features in the part.

A methodology for automatic recognition of features from a CAD model is presented here to deal with these issues. It is based on the Backward Growing strategy [Wang 93], which builds the stock from a part by adding on features at each step. However, we have introduced several enhancements which enable us to deal with the above issues, as well as to overcome machinability problems associated with Wang's method [Chamberlain 93].

2. FEATURE REPRESENTATION AND SYSTEM ARCHITECTURE

The *delta* volume ($\Delta = Stock -^{*} Part$) can be decomposed into a union of a set of manufacturing features, i.e. $\Delta = \cup^{*}_{(F_i \in F)} F_i$, where F_i is the feature instance belonging to feature class F. The motivation of this research is to generate a reasonable minimum set of machinable feature instances to fill in the delta volume Δ.

Figure 1 shows a classification of the features used by us They are divided into two classes: those with fixed geometric elements (only the parameters vary, between different feature instances), are called template features; the other class is a general profile pocket.

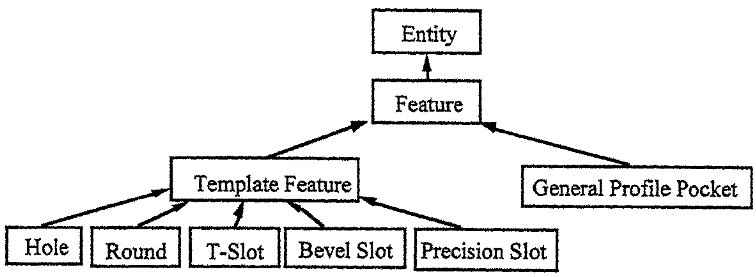


Figure 1. The Feature Class Hierarchy

Template features are defined to meet the features with special form characteristics such as holes, T-Slots, rounds, and precision slots. Precision slots are slots which need to meet specified alignment requirement or precision constraints. The other class, general profile pocket features are used to describe most of the delta volume. The general profile pocket is generated by extruding a 2D planar shape by a fixed distance (the depth of the pocket) along the normal to the plane of the shape. The shape is defined by a simple, closed outer boundary, and a (possibly empty) set of pairwise non-intersecting islands contained fully within the outer boundary. The boundary as well as the islands are defined by ordered sets of directed edges. Each edge has a geometry specified by either a straight line or a circular arc. Each edge is known to be either virtual (violable), or machined (non-violable).

The architecture of the proposed feature recognition system is shown in Figure 2. The input consists of the part BRep, material feature information and precision feature information such as datums, tolerances, attributes, and possibly the design form features information.

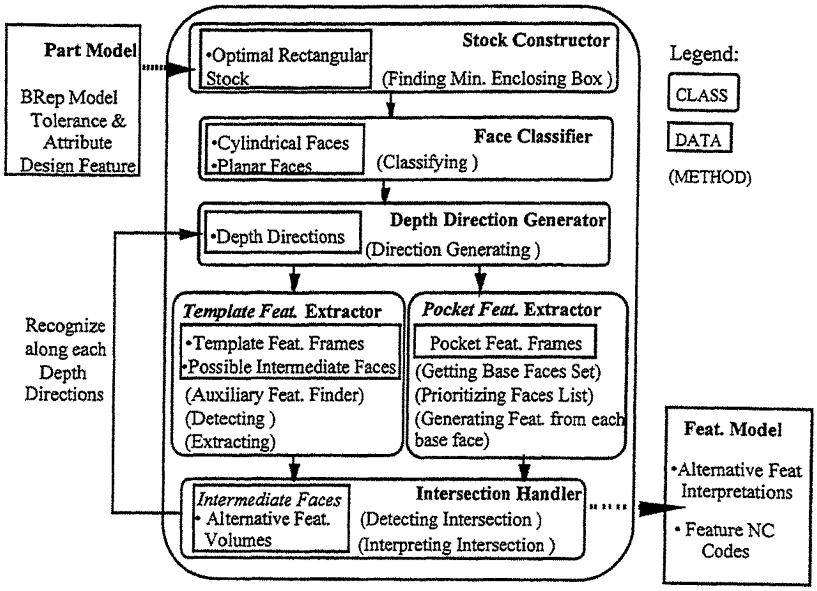


Figure 2. Architecture of the Feature Recognition System

The first step is to generate the minimum box cover around the solid. This serves two purposes: it minimizes the material removal volume, and it makes the recognition process invariant with respect to coordinate transformations of the part. The next step is the classification of face types to identify template feature faces. *Depth directions* are orthogonal directions along which general profile pockets can be grown. The depth direction generator identifies and sequences these directions. It also identifies and marks some template features, isolated wedge shaped pockets, and performs splitting of orthogonal faces with obstructions along a given depth direction. Information such as datums or analyzing face's characteristics are also utilized by the depth direction generator class to prioritize the candidate depth directions of the part model. Depth direction lists are used to sequence the feature extraction. After getting a potential feature, its interaction with other features is analyzed by detecting the completeness of bounding faces. Based on the *intermediate faces* information output in feature extraction procedure, the intersection handler class will deal with the intermediate faces and derive the alternative feature interpretations. The following section gives the details of some of these methods.

3. RECOGNITION METHODOLOGY

The implementation of the minimum box works in two steps. First, working from a polygonal approximation of the part, it's convex hull is generated. An extension of

O'Rourke's [O'Rourke 85] method of double rotating calipers which can determine the minimum enclosing box of the convex hull is then applied.

Feature recognition takes place in phases. We first classify each face on the part according to its geometry and orientation. This clues the system about the possible role of the face (whether it will be a part of a template feature, and if so, which kind).

Faces related to template features are eliminated at the outset, by filling unintersected holes and wedges. The remaining faces are examined for growing using general profile pockets. Pockets are grown using *depth-directions*. Depth directions are orthogonal directions along which pockets can be extruded. The area of each face that can be grown along each depth direction is evaluated (that is, the approachable area). Next, the faces associated with each depth direction are prioritized "from deep to shallow." Heuristics are used to sequence the depth directions. The approachable area of the deepest growing base face along a depth direction is extruded upward to form a general profile pocket. An extension of Wang's technique is used to determine the extrusion thickness. This, as in the case of Wang, grows the part in the most conservative steps. Pocket extraction continues along a given depth-direction till the growing face reaches the bounding box. Then the next depth direction is examined.

Once the delta volume has been filled up with features, feature interactions are examined. We introduce the concept of *intermediate face* to handle the feature intersection detection and the corresponding alternative feature representations. *Intermediate faces* are easily detected from the partially destroyed feature face set. We can also deduce possible alternate *intermediate faces* for one particular intersection, which can easily be mapped to the alternative feature representations for the part (example shown in Figure 3).

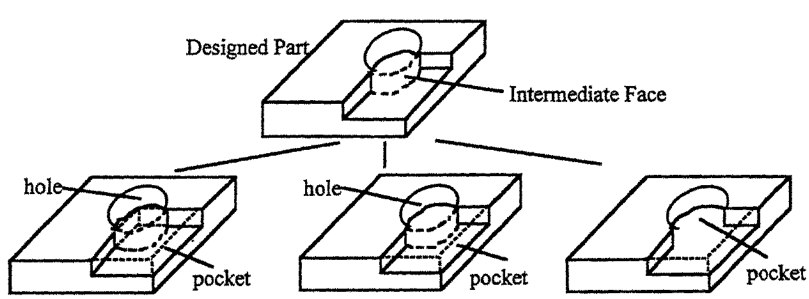


Figure 3. Deriving Alternative Interpretations from Intermediate Faces

The best of these three interpretations depends on the geometric parameters, the tools sizes (and the fixtures) available. The recognition module generates potentially good alternatives, and submits these interpretations to the process planning functions. The modules of the integrated process planning system are introduced in the following section.

4. INTEGRATED PROCESS PLANNING SYSTEM ARCHITECTURE

The recognition methods described above form a part of an integrated process planning system, called OSCAP (Open Architecture System for Computer Aided Process Planning) being developed currently at the HKUST. The idea of OSCAP is to develop a process planner which can easily be expanded or shrunk in terms of the functions it provides to its user. Figure 4 shows the overall structure of OSCAP. Each module attached to the OSCAP core provides some functions about which the core maintains information. When any module requires some function, it requests this service from the core, which in turn can use any of the other modules attached to it at that time. All information is exchanged between modules in STEP (now ISO 10303) compliant format.

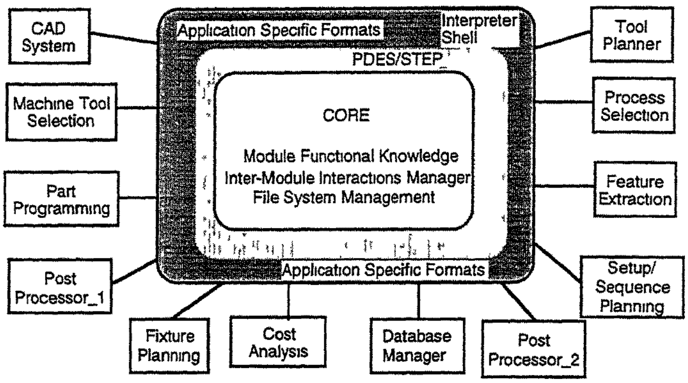


Figure 4. The OSCAP architecture

The solid modeler used for OSCAP is ACIS™. The fixture and setup planners are adapted from earlier research work by one of the authors [Joneja 94]. Figure 5 shows the architecture of the modules described in this paper. The part programming module includes a powerful pocket machining system, capable of generating machining plans for arbitrary shaped pockets with islands and holes.

An important feature of OSCAP is its emphasis on open exchange of information between different functions. There have been two major obstacles in the widespread use of generative process planning in the modern industry: first, the restrictions placed by weak feature recognition procedures made it difficult to justify the expense of using automation at this level. Second, with the boom of CAD software in the last decade, many companies are locked in to specific CAD products; most process planning systems are developed with integrated CAD packages, which make it impossible for companies to adopt CAPP unless they also use the same CAD system. With the use of STEP to communicate the product and manufacturing related information, OSCAP aims to eliminate the latter of these obstacles. We also believe that the feature recognition methods introduced in our work are powerful enough to handle reasonably complex, practical parts.

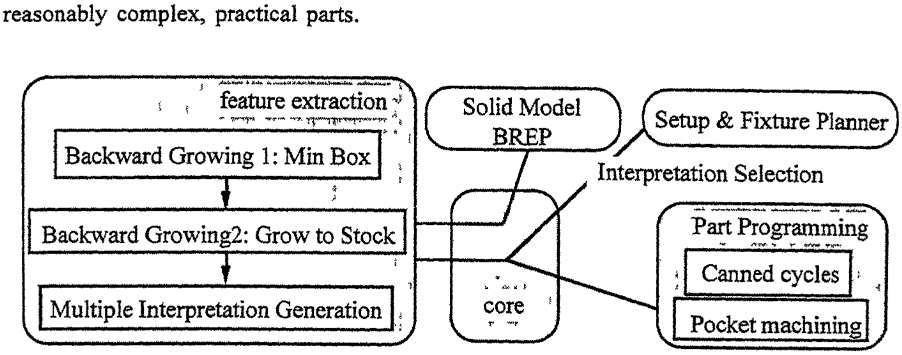


Figure 5. The Interaction of the Feature Recognition and Process Planning Modules

5. REFERENCES

Chamberlain, M A., Joneja, A., Chang, T. C., 1993, "Protrusion Features Handling in Design and Manufacturing Planning," *Computer-Aided Design Journal*, vol. 25, no. 1, pp. 19-28

Cutkosky, M. R., Tenenbaum, J. M., "A Methodology and Computational Framework for Concurrent Product and Process Design", *Mech. and Machine Design*, vol. 23, no. 5, 1990.

Gavankar, P. S., Henderson, M. R., "Graph-Based Extraction of Two-Connected Morphological Features from BReps," *J of Intelligent Manufacturing*, (1995) 6, pp. 401-413

Gupta, S. K., Kramer, T. R., Nau, D. S.,et al, "Building MRSEV Models for CAM Applications," *Advances in Engineering Software*, 1994, 20 (1994), pp. 121-139

Henderson, M. R., "Extraction of feature information from three-dimensional CAD data", PhD thesis, Purdue University, USA, 1984

Joneja, A., Chang T. C., 1994, "Setup and Fixture Planning in Automated Process Planning Systems," submitted for publication in *IIE Transactions*

Joshi, S., Chang, T. C., "Graph Based Heuristics for Recognition of Machined Features from a 3D Solid Model," *Computer-Aided Design*, March 1988, 20(2), pp. 58-66

Karinthi, R. and Nau, D., 1991, "Geometric Reasoning for Design and Process Planning," *Procs of the 1991 NSF Design and Manufacturing Systems Conference*, Austin, pp. 497-502

Kim, Y. S, "Recognition of Form Features using Convex Decomposition," *Computer Aided Design*, September, 1992, 24(9), pp. 461-476

O'Rourke, J, "Finding Minimal Enclosing Boxes," *International Journal of Computer and Information Sciences*, vol 14, no 3, 1985, pp. 183-199

Vandenbrande, J. H., Requicha, A. A. G., "Spatial Reasoning for Automatic Recognition of Machinable Features in Solid Models," *IEEE transactions of Pattern Analysis and Machine Intelligence*, December 1993, 15(12), pp. 1269-1285

Wang, M. T. et al, "Manufacturing Feature Extraction and Machined Volume Decomposition in a Computer-Integrated Feature-Based Design and Manufacturing Planning Environment," *Computers in Industry*, 23, 1993, pp. 75-86

Woo, T. C., "Feature Extraction by Volume Decomposition," *Procs. Conference on CAD/CAM Technology in Mechanical Engineering*, March 1982, pp. 76-94

FEATURE RECOGNITION FOR NC PART PROGRAMMING

R. Tuttle*, G. Little*, D. E. R. Clark**, J. Corney*
*Department of Mechanical & Chemical Engineering, **Department of Mathematics
Heriot-Watt University
Riccarton Campus
Edinburgh - SCOTLAND
EH14 4AS

ABSTRACT

A recently developed system for automatically generating machinable volumes from solid models of multi-sided components and their stock is described. The volumes are used to support the generation of Numerical Control (NC) machine code from a commercial solids-based machining package. Unlike much of the reported work in this area, the goal has not been to produce an entirely automated system, but rather that of an intelligent assistant, capable of increasing the part programmer's productivity. The methodology demonstrates one way in which the twin problems of feature interaction and multiple interpretations can be side-stepped. The system generates machinable feature volumes bounded by planar and cylindrical faces using algorithms which have been developed and considerably extended from previously reported work. While the system is capable of stand-alone feature recognition using its own Graphical User Interface (GUI), it has also been integrated into the commercial solids manufacturing package, SolidCAM[1].

KEYWORDS

Feature Recognition, Cycles, Feature Volumes, Manufacturing, Multi-sided

1. INTRODUCTION

Automated recognition of features, based upon Boundary Representation (B-Rep) solid models, has been developing for over two decades, evolving from seminal work by Kyprianou (1) on shape classification in Computer Aided Design (CAD). He showed that a loop of *convex* edges on a solid model indicated the presence of a depression feature, whilst a loop of *concave* edges *signalled* a protrusion feature.

Since then a number of researchers have developed a range of methods of feature recognition each tailored to a particular application. Two of the more productive areas of research are the volumetric and the graph-theoretic approaches. The former approach has been developed typically in the work of Kim et al (2) and Sakurai (3, 4), whilst the latter method has been brought to fruition in the work of such researchers as DeFloriani (5) and Sormaz et al (6). Almost without exception, these feature recognition systems identify the features as *face sets*. The methods for achieving this vary dramatically but the outcome is essentially the same - namely a feature either wholly or partly bounded by a closed cycle of faces and/or edges on the component.

Corney and Clark (7) used a graph-based methodology to view solid models with respect to a specific machining direction. Their work was bound by the assumption that the component be single-sided, i.e. the component may be completely machined from one tool approach direction. This work

[1] SolidCAM is the copyright of CADCENTRE Ltd., Cambridge, UK

has since been extended by Tuttle et al (8) encompassing increasingly complex geometry for applications in both computer aided manufacturing (CAM) and numerical control (NC) machining.

It would appear that to date very little of this plethora of feature recognition research has, until recently, left the confines of research establishments and literature. Recently however, several commercial products have appeared which are either self-contained applications based on feature recognition from pre-defined libraries (such as the PART (9) system for process planning) or are capable of identifying user-defined patterns (templates) of geometry and topology in isolation from any particular application (10). These types of technology also offer methods of linking information about a component shape with downstream activities such as cutter path generation or process planning and manufacturing schedules.

2. THE FEATURE RECOGNITION ALGORITHM

The generic recognition algorithms previously developed by the authors for polyhedral objects (7) have been significantly extended to enable the interrogation of more complex geometry and the recognition of cycles associated with both 'closed' and 'open' manufacturing features. Once identified, these feature profiles are extruded to form a set of discrete 3D volumes (the *delta-volumes*) which are selected as material to be removed from the original stock for component manufacture. These volumes are then passed on to a commercial solids machining package. The feature recognition algorithms are now capable of feature recognition on *multi-sided* components (i.e. components requiring more than one machining direction for manufacture).

A specially designed Graphical User Interface (GUI) operates between and/or within a manually driven solids machining package and the automated feature recognition system. This reflects the desire to create an automatic recognition system which can *assist* the production engineer in the identification of volumes for removal from stock in order to manufacture a component. The interface allows visual interpretation the manufacturing features from the geometry of the component(s). The user can keep track of the default (or user defined) volumes created following the initial feature recognition stage of the process. The amount of remaining stock at each stage of volume creation can also be monitored, if required.

2.1 Algorithm Description

The feature recognition algorithm was originally developed from a technique described in Corney and Clark (7), which had been developed to interpret the geometry of polyhedral single-sided 2.5D components. However, the original algorithm was seen to be inefficient because many of the geometric interrogations on components were repeated, resulting in a large amount of backtracking. Consequently, a new approach was developed in which the interrogations are performed prior to the search algorithm.

Furthermore, the extended algorithm is capable of handling both planar and cylindrical faces and, moreover, will output feature volumes. Yet another facet of the refined algorithm is that it collects geometric information on any interacting features (e.g. *nested* depressions). The default depression volumes created then, by their very nature, preclude the formation of any intersecting volumes as these are normally unnecessary for component manufacture. A detailed description of the refined algorithm and search methodology is documented elsewhere (6) a brief description only is included here.

The operational procedure for the algorithm is in three stages :-

1. The path generation phase : in which each vertical face (relative to the machining direction) is considered in isolation. Possible paths across the face (travelling from left to right) are generated and recorded, and a directed graph is created.

2. Cycle identification phase : in which the directed graph constructed in the first step is analysed to find cycles of faces bounding depressions. Then, using the orientation of the faces, the closed cycles of faces found during the search are identified as protrusion, depression or open features.

3. Generation of Feature Volumes: in which individual face cycles are used to construct a 2D profile. This profile can then be swept to a height which is either user defined (via the selection of edges) or generated automatically by the default procedure.

An important aspect of this process is that in each step heuristics are employed to reduce the number of paths and cycles generated. The result is an efficient, robust recognition system. The depression features can be thought of exclusively as *volumes of material which have to be removed from the stock during manufacture*. To transform these face cycle features into volumes, the code creates a 2D profile of each cycle and then analyses the relative heights of edges in the cycle. Default volume creation specifically precludes the generation of *redundant* intersecting volumes. In this way, the refined recognition algorithm fulfils the production requirements of creating machinable feature volumes without the inherent restrictions of a feature library.

The algorithm has a number of efficiency gains over other comparable feature recognition systems:

◊ No pattern matching or sub-graph comparisons are necessary for features (i.e. no feature library requirement).
◊ Features are found and feature volumes are created directly from B-Rep data extraction.
◊ Heuristic approach without any complex rule-base.

2.1.1 For Multi-sided Component Feature Recognition

For feature recognition on multi-sided components a number of additional changes have been made to the algorithm, though its basic feature recognition procedure remains unaltered from that detailed above. These changes are now outlined.

In order to facilitate the analysis of multi-sided components further improvements have been made to the original algorithm in the form of an 'approachability' check. A multi-sided component can be defined as a component whose manufacture requires machining from two or more directions. In previous work by the authors on single-sided components (6, 7, 8) only one *aspect* (or *machining approach) direction* was applicable. For multi-sided components, however, a number of aspect directions are required. The actual number of aspects will vary depending upon the profile of the component but each aspect can be specified by a unit vector anti-parallel to a planar face on the component.

When analysing a multi-sided component, as each aspect direction is selected for feature recognition, the valid paths are found as before, therefore *without* any check on visibility of the path from that machining direction. The visibility, or *approachability*, of the path is then determined by analysis of its start and end bounding points. If a human observer was to determine the visibility of a point on a component from a specific direction (i.e. *a machining approach direction*) then a primary indicator of visibility of the point would be that light can travel directly from the said point to the human eye *without obstruction*. In a similar manner to a ray of light, a solid modeller 'ray' is used to detect path visibility from any given aspect direction. The ray is specified to travel along a specific path (*the machining path*) in a given direction (*opposite to the aspect direction*) from a given point (*start and end points of the path in question*) and a record is kept of any entities it encounters *during its journey*.

By firing a ray from the start and end points of a path, a check can therefore be made of any ray interactions with entities of the component. If a ray passes the outer profiles of the body uninterrupted then it is labelled as *approachable*. Once all paths have been labelled as approachable or not the feature recognition algorithm continues to the cycle generation stage. Only paths which have been deemed approachable are used for cycle generation and machining volume creation, as previously described.

3. INTEGRATION WITH SOLIDCAM

The FeatureFinder is capable of stand-alone operation in its own GUI. For single-sided components the operation of this system has been documented (8). The major difference for a multi-sided operation is that a succession of aspect directions may be selected. In such a modus operandi the volumes created have *attached attributes* detailing the aspect direction from which they were found/visible. They are then saved in an ACIS[2] file format and, as such, can be imported and manipulated as required by the user (perhaps using other proprietary software). This feature recognition system has also recently been integrated into the solids manufacturing package SolidCAM for use in assisting the generation of NC part programs for machine tools in CAM.

The integrated version automatically passes any volumes created to the manufacturing engineer for attachment of machining and process planning information. Within this version it is not necessary to specify the aspect direction from which to search for manufacturing features since this is, by default, the currently active co-ordinate system within the SolidCAM package. This greatly simplifies operation for the end user. Screen dumps of the integrated version of SolidCAM are shown in figures 1-3.

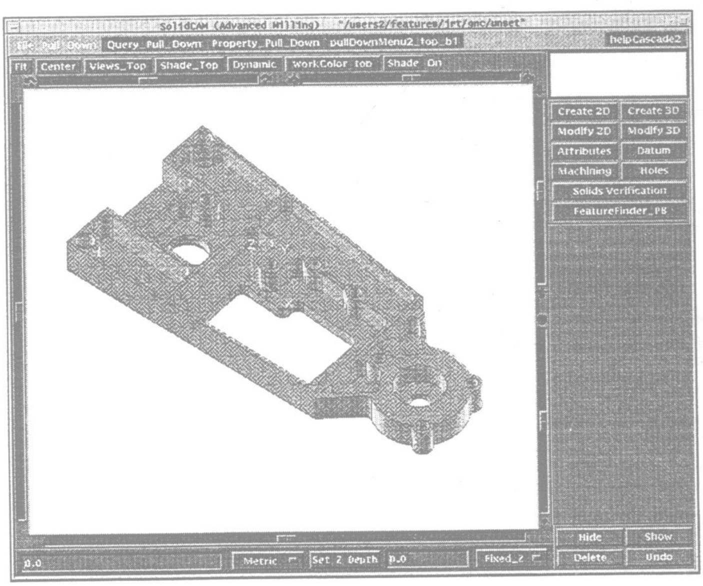


Figure 1. SolidCAM with Component Loaded

[2] ACIS is the copyright of Spatial Technology Inc., Boulder, USA.

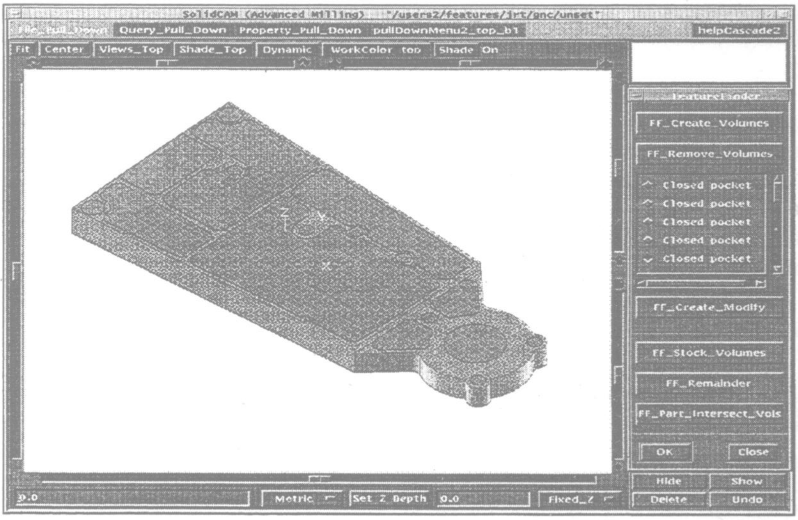


Figure 2. SolidCAM with Feature Recognition and Created Volumes

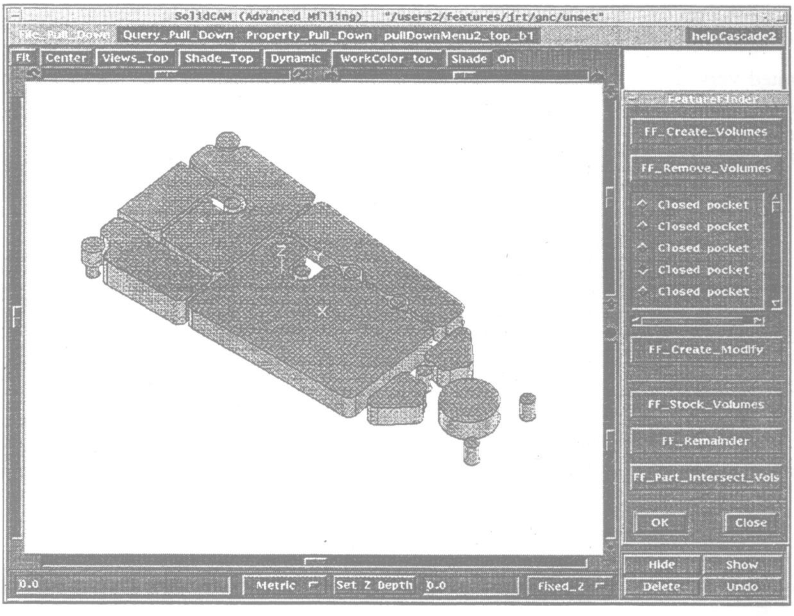


Figure 3. Volumes Automatically Created using FeatureFinder

4. CONCLUSIONS

◊ A feature recognition algorithm has been described which is capable of interrogating the geometry and topology of a wide range of multi-sided components and of generating delta-volumes for manufacturing with information on tool approach directions attached.

◊ A prototype operational feature recognition user interface has been described. The package is capable of interfacing with a commercial CAM package and communicating information on component manufacture.

◊ The FeatureFinder software has been integrated into the commercial CAM package SolidCAM and, in this format, has a seamless mode of operation for exchange of manufacturing volumes.

◊ Work is ongoing to increase the functionality of the software and user interfaces to further aid manufacturing and process planning.

5. ACKNOWLEDGEMENTS

The authors wish to thank the collaborating companies, MacTaggart, Scott & Co Ltd, Edinburgh, UK and CADCentre Ltd, Cambridge, UK for their assistance and part funding of the project. We would also like to thank the Engineering and Physical Sciences Research Council (EPSRC) for their continued financial support of the research under grant no. GK/K48020.

6. REFERENCES

1. Kyprianou L. K., 'Shape Classification in Computer-Aided Design,' PhD Thesis, Christ's College, University of Cambridge, 1980.

2. Kim Y. S., Roe K. D., 'Conversions in form feature recognition using complex decomposition.' Proceedings of 1992 ASME International Computers in Engineering Conference, Aug 2-6, San Francisco, CA, pp.233-244, 1992.

3. Sakurai H., Gossard D., 'Recognizing shape features in solid models.' IEEE Computer Graphics and Application, September 1990, Vol 10, No5, pp.22-32, 1990.

4. Sakurai H., Chin C.W., 'Definition and recognition of volume features for process planning', Advances in Feature Based Manufacturing, Elsevier/North Holland, 1994.

5. DeFloriani L., 'Feature Extraction from Boundary Models of Three-Dimensional Objects', IEEE Transactions on Pattern and Machine Intelligence, Vol 11 No 8, August 1989, pp.785-798, 1989.

6. Sormaz N., Clark D.E.R., Corney J., Tuttle J. R., 'A Feature Recognition Algorithm for NC-Machining.' Proceedings of The 5th IFIP Workshop, WG5.2, Workshop on Geometric Modeling in CAD, May 1996, Airlee, Virginia, 1996.

7. Corney J., Clark D.E.R., 'Efficient Face-Based Feature Recognition', Proceedings of Second Symposium on Solid Modeling and Applications, ACM Press 1993, pp.313-322, 1993.

8. Tuttle R., Sormaz N., Clark D.E.R., Corney J.R., 'Integrating Solids-Based Machining with Feature Recognition'. Proceedings of 1996 ASME Design Engineering Technical Conferences and Computers in Engineering Conference, August 18-22, 1996, Irvine, CA, 1996.

9. PART White Paper, ICME Technologies, 1994

10. Risbud A., 'Automatic Feature Recognition', Spatial Relations, Vol 12, October 1995, Spatial Technology Inc, Boulder, Colorado, pp 3-4, 1995.

ASSEMBLY LINE BALANCING USING MODIFIED GENETIC ALGORITHMS

Y. GUO* and K.C. CHAN
Department of Mechanical Engineering
The University of Newcastle

ABSTRACT:

This paper presents a modified genetic algorithm (GA) to solve assembly line balancing (ALB) problems. Two modifications are made and a new notion is introduced in the proposed GA. The first modification is to apply the *elitist strategy* in reproduction. The other is to replace the less fitting solutions with the children instead of replacing their parents. And in order to differentiate between the solutions in the simple ALB problem, the notion of "equal pile problem" is introduced. A number of examples, covering the simple ALB problem and multi-objective ALB problem, are presented with better results than the ones with the other methodologies.

KEYWORDS

Genetic Algorithms, Assembly Line Balancing, Elitist Strategy, Equal Piles Problem

1. INTRODUCTION

Assembly line balancing (ALB) is known as one of the most difficult combinatorial optimisation problems and has been solved using linear programming and dynamic programming. Unfortunately these approaches do not lead to efficient algorithms [1]. Genetic algorithms (GAs) are powerful solvers of optimisation problems and have been successfully applied to various optimisation problems such as the travelling salesman, scheduling and partitioning objects. However, in order to successfully apply GAs to ALB problems, a suitable algorithm has to be elaborated and even modifications to the conventional GAs are necessary. The purpose of this paper is to generate a suitable genetic algorithm for ALB.

2. PROPOSED GENETIC ALGORITHM

2.1 Modifications in Proposed Genetic Algorithm

A few reports related to application of GAs to ALB are found with some problems such as that the maximum fitness reduces with generation number [1,2,3]. To generate a good genetic algorithm, the special emphasises are placed upon the following aspects.

Firstly, we apply the *elitist strategy* in the proposed GA to keep the best fitting solution of each generation for the next. The number of *elitist* is set to one. It is found experimentally that commonly used reproduction does not always select the best fitting solution of each generation for the next operation and if the best solution is not selected, for ALB problems it may take many generations or forever to reproduce. For the GAs for ALB, the *elitist strategy* is only applied by Kim et al [2].

Next, in the crossover operation, we replace the less fitting solutions with the children generated by the parents which have higher fitness values. We found that for ALB problems, if the parents are replaced by their children, the maximum fitness in each generation will reduce with generation. Such trends can be found in figure 2 of [2] and little improvement of the maximum fitness with generation is reported in figure 2 of [1] although a number of different crossover operators were

* Corresponding author

applied. Leu et al.[3] use the "elitism" replacement approach to avoid this problem and improvements with the generations were achieved, but their population increases with the generation. In our proposed GA, single point crossover operator is selected to perform crossover operation, but we replace the less fitting solutions with children in each generation. In this way, the population is unchanged with the generation. For the ALB problem, the replacement approach in the crossover operation is critical for the application of GA for the reason given bellow:

When the tasks are assigned to the work stations and the objective is to minimise the number of the stations or minimise the multi-objective, the most important factor in deciding the fitness value is the number of stations. However, in the initial population, the number of solutions that have the same number of stations is very small and may be just one or two. Moreover, the number of stations for all potential solutions in a generation is different. Therefore, in the crossover operation, the chance of selecting two parents for crossover with the same number of stations is very small. So, after crossover operation, the number of stations of children very often is the number of stations of the parent that has the bigger number of stations. The result is that the children have worse fitness values than the one of the parents because of the bigger number of stations. If the normal replacement approach is used, (replace the parents with the children) a parent with a better fitness value is replaced by a child with a worse fitness value. The Figure 1 is the plot of average of maximum fitness generated by both the commonly used GA and proposed GA for the Tong's 70 task problem with the objective of the minimisation of the number of stations. The both GAs use the same parameter setting in order to compare the results.

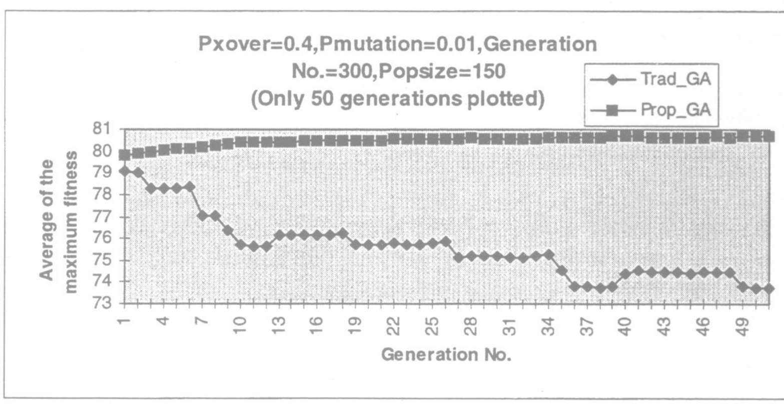


Figure 1
(Comparison of conventional GA & proposed GA)

Besides the crossover operation, commonly used mutation is also modified such that the genes selected will be mutated within the number of stations. In this way, the chance to generate a feasible solution is increased.

The notion of "equal pile problem" is introduced to the proposed GA to differentiate the optimum solutions for the simple ALB problem. For the simple ALB problem, the proposed GA without this introduction found many solutions that have the same minimum theoretical number of stations but different assignments of tasks. The introduction of the notion of "equal pile problem"[4] is used to find the solution that most equally assigns the tasks to the stations. For the ALB problem, it is derived as follows.

$$D = \sum | T_k - T_a | \quad (3)$$

D is the total processing time difference when the summation is over all stations, T_k is the total processing time for the station k and T_a is the average T_k. Therefore, D can be used to show how equally tasks are assigned to the all stations. With the same number of station, a solution with a bigger D shows the tasks more unequally assigned to the stations.

To combine the total processing time difference into the objective, the following additive utility function is employed.

$$U = a N + b D \quad (4)$$

Where U is the objective utility function that combines two sub-objectives together and N is the number of stations. 'a' and 'b' are two coefficients. The values should be chosen to show the weights of two sub-objectives. For such ALB problem, the key objective is to minimise the number of stations (N) while the total processing time difference D is only used if several solutions with the same number of stations are found. Therefore, the weight of N is much bigger than D and the U value for a potential solution with a larger number of stations will be much bigger than the U value for a potential solution with a smaller number of stations. The value of 'a' and 'b' is decided only based on the consideration of such weight.

2.2 Other Aspects

We use Group-Number Encoding method to represent the potential solutions, by which the number k in the position i indicates that the task i is assigned to the station k. Two different methods to initialise the population are also studied in this paper. The first one is the random assignment of tasks to stations. The second is called COMSOL, developed by A. L. Arcus in 1966 [5].

In the proposed GA, the fitness function have an inverse correlation with the objective functions since the objective function in ALB problem is to minimise the number of stations or minimise the cycle time or minimise multi-objectives. Two different fitness functions were chosen for the different type of ALB.

Fitness function 1: $Y = 10^5 * 2^{-\log(x)}$ (5)

Fitness function 2: $Y = 10^3 * X^{-\log(X)}$ (6)

Control parameters, such as population size, generation number, crossover probability, mutation probability, are set in the following way in the proposed GA. We initially set the range of crossover probability from 0.1 to 0.99 with the step 0.1, the mutation probability from 0.005 to 0.035 with the step 0.005, the generation number from 5 to 500 and population size from 10 to 300. After the initial trials, we narrow the range of these parameters and then varied one of the parameters with others fixed. For each choice of parameters, we run the program 20 times to locate the best value of the varied parameter in terms of counts of the optimum solution in all trials. Then using this best value with others fixed we located the best value of another parameter by varying its value. Using the same method, the best values for the other parameters are also located.

Rejection method is used to treat infeasible solutions in both population initialisation and genetic operations.

3. EXPERIMENTS

The proposed GA described above is implemented in the Borland C programming language. The experiments to be undertaken come from three famous predefined ALB problems. They are: Jackson's 11 tasks problem, Mitchell's 21 tasks problem and Tonge's 70 tasks problem. With different objectives, these three problems become the following five experiments. The results are shown in the table 1.

Experiment	Objective	Utility Function	GA Optimum Solution	Results By other method	Notes
Jackson's 11 Task problem	Multi-objective (f_1,f_2,f_3)	$U = -40*(f_1 - 1)^2 - (f_2 - 7)^2 - 10^{-6} * (f_3 - 27840)^2$	U = - 2849 (N=3,CT=18, Cost=11896)	U* = - 2855 (N=3,CT=16,Cost=13230)	Same utility function as [6], The GA found a better result.
Jackson's 11 Task problem	Minimise Cycle Time (CT) given N=5,	No utility function used	CT=10	CT=10	GA found four different assignments of tasks with the same CT
Jackson's 11 Task problem	Minimise N given CT=10	U = N + 0.5 * D	N=5 with D=3.2	N=5 with D=4.8	GA found two different assignments with a smaller D than [7]]
Mitchell's 21 task problem	Minimise N given cycle time (CT)=20	U=10*N + D/6	N=6 with D=2	N=6 with D=9	GA found three more solutions with same N and smaller D than [8]
Tonge's 70 task problem	Minimise N given CT= 364	U=N+0.01*D / 11	N=11 with D=30.9	N=11	GA found many more solutions with same N.

Note: In [5], f_1 is the objective one, minimise the number of station (N). f_2 is the objective two, minimise the cycle time. f_3 is the objective three, minimise the operation-short-term costs.

Table 1: Summary of Experiments

The utility function is used to combine different subobjectives. The one used in the first experiment is the same as in [6] in order to compare the performance of the proposed GA with [6]. The others are only used to differentiate the optimum solutions for the first type simple ALB problem. For the first three experiments, the random assignments of the tasks to stations are used to initialise the population whereas for the rest, COMSOL is used.

Figure 2 to 5 are plots of average of the maximum fitness over generation number for the first four examples listed in the table 1. The plot of the last example is already shown in the figure 1. From these figures, we can see the improvement of the maximum fitness over generation number. Kim et al, Leu et al and others have already shown that the application of GAs to ALB problem is more efficient than other methodologies. The proposed GA to ALB explore further the advantage of application of GA to ALB. The results given in the last three columns have shown this. For the first experiment, a better solution was found with the proposed GA in terms of utility function. In searching the optimum solution, one more better solution was also found, which has a utility function U = -2845 (N=3, CT=17, Cost=12537). For the second experiment, the proposed GA found four solutions with the same number of stations. For the last three experiments, because of the introduction of the notion of "equal piles problem", the proposed GA is able to differentiate between the solutions with the same number of stations and therefore found the one that most equally assigns the tasks to the stations. All experiments show that the proposed GA can provide much better solutions than the other methodologies in [6][7][8].

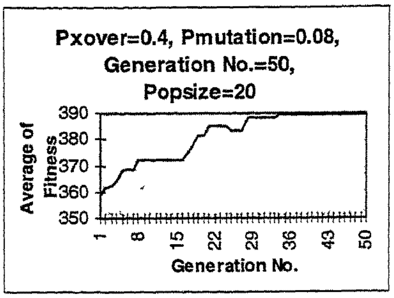


Figure 2
(Plot for Jackson's 11 task with multiobjective)

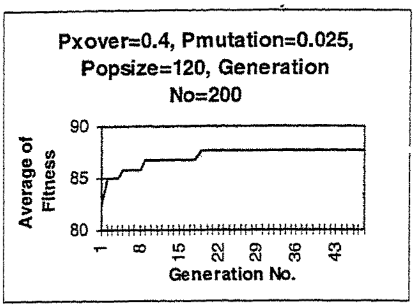


Figure 3
(Plot for Jackson's 11 task with minimisation of cycle time)

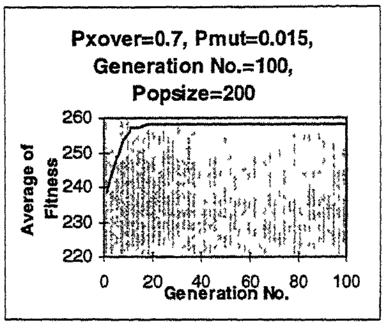


Figure 3
(Plot for Jackson 11 task with minimisation of number of stations)

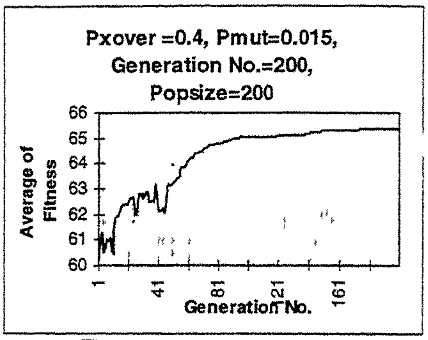


Figure 4
(Plot for Mitchell's 21 task minimisation of number of stations)

4 SUMMARY AND CONCLUSIONS

This paper presents a genetic algorithm to solve ALB problems. The proposed GA is different from all genetic algorithms proposed by other researchers and performs very well. Firstly we use "*elitist strategy*" in reproduction and replace less fitting solutions with children in every generation. And then we introduce the notion of "equal pile problem" to differentiate between solutions. Experiments show that it is effective and can find the optimal solution. The replacement approach in the crossover operation is critical to the performance of the GA for ALB. Replacing the less fitting chromosomes with children is the best way to improve the maximum fitness over generation. "*Elitist strategy*" applied in the reproduction can also improve GA performance because it can avoid disappearing of the best solution of a generation in the reproduction. The notion of "equal piles problem" should be introduced to differentiate between the solutions because the most ALB problems do not have unique solution and it can find the one that most equally assigns the tasks to the stations.

5. REFERENCES

1. Tsujimura Y., Gen M., Kubota E. "Solving fuzzy Assembly-line Balancing problem with genetic algorithms", Computers & Industrial Engineering. Vol. 29: 543-547, 1995

2. Kim Y. K., Kim Y. J., Kim, Y. "Genetic Algorithms For Assembly Line Balancing With Various Objectives". Computers and Engineering Science, Vol. 30, N0.3, pp. 397-409, 1996.

3. Leu Y.Y., Matheson LA., Rees LP. "Assembly line Balancing using algorithms with heuristic-generated initial populations and multiple evaluation criteria", Decision Sciences. 25(4): 581-606, Jul-Aug. 1994.

4. Donald R. Jones and Mark A. Beltramo, " Solving Partitioning Problems with Genetic Algorithms", Proceedings of the Fourth International Conference on Genetic Algorithms, Morgan Kaufmann Publishers, Los Altos, CA, pp. 442-449, 1991.

5. Arcus, A.L "COMSOAL: A Computer Method of Sequencing Operations for Assembly Lines", Operation Research, Vol. 4 No.4, 1966

6. Malakooti B. "A Multiple Criteria Decision Making Approach for the ALB Problem" International Journal: Production Research, Vol. 29, No. 10, pp1979-2001, 1991.

7. James R. Jackson " A Computing Procedure For A Line Balancing Problem" Management Sciences, 2(3), pp261-271 (1956)

8. Fred M. Tonge, "Summary of a Heuristic Line Balancing Procedure" Management Sciences, 8(1), pp 21-39 (1960)

AGENTS AND MULTIAGENT MANUFACTURING SYSTEMS: MODELLING SUPPORT

Edward Szczerbicki
Department of Mechanical Engineering
The University of Newcastle, Newcastle, NSW Australia

ABSTRACT

The problem of modelling of an information flow in manufacturing agents (subsystems) consisting of people, machines, robots, etc., matches very frequent real-life situations in which the following should be answered: "How to structure an exchange of information between the elements of an agent? What is better, complete information but heavily delayed, or incomplete information less delayed?" The formal quantitative model can be used to generate some examples of knowledge useful in answering the above type of questions. Quantitative models of an information flow evaluation, however, are often too complex to serve as the tools useful in the knowledge retrieval process. In the paper, the application of qualitative simulation in the process of knowledge acquisition for manufacturing agents is investigated and illustrated with examples.

KEYWORDS

Manufacturing Agents, Qualitative Modelling, Decision Trees, Signed Directed Graphs

1. INTRODUCTION

A quantitative model can be helpful in creation of knowledge connected with an information flow evaluation in autonomous manufacturing systems, where autonomy is defined as information closure [1]. Because of its complexity the model cannot be used for analysis and evaluation of an information flow in all possible decision situations faced by an agent. Qualitative modeling and reasoning, on the other hand, are areas of Artificial Intelligence (AI) that focus on reasoning about the behaviour of real life complex systems without relying on numbers [2]. The underlying idea in qualitative modeling is that the complexity of a given system can often be reduced by taking into account only certain abstractions of its behaviour [3]. In quantitative modeling the idea is to represent the system as a set of parameters that can assume real values and mimic the functioning of the system by the set of changes of these parameters in time.

What follows is a short quantitative description of an informational model of a manufacturing agent. Next, two different non-quantitative tools are proposed for addressing the problem of knowledge acquisition for an autonomous agent in various decision situations.

2. INFORMATION FLOW FOR AUTONOMOUS AGENTS: PRINCIPLES AND TOOLS FOR QUANTITATIVE MODELLING AND EVALUATION

Let **A** represent the set of possible actions which can be undertaken by the element of an agent, **Z** the set of corresponding consequences, and **X** random variables describing the actual state of the external environment. It can be assumed that $z=f(a,x)$ as the particular consequence (z) depends usually on an action (a) undertaken in the particular state of the environment (x). On the other hand, the decision about particular action depends on information that is available about the state of the environment. If ß stands for the decision function, we have $a=ß(d)$ where d represents information. It can be shown that, after considering certain correlation between information, action, and energy, we have [1]:

$$f(a,x)=\mathbf{B}_0 - 2\mathbf{B}^T\mathbf{A} + \mathbf{A}^T\mathbf{Q}\mathbf{A}, \qquad (1)$$

where $\mathbf{B}_0=b_0(x)$, $\mathbf{A}=[a_i]$, $\mathbf{B}=[b_i(x)]$, symmetric matrix $\mathbf{Q}=[q_{ij}]$ ($i,j=1, 2, ..., n$) and n represents the number of elements of an agent. Minimum of (1) exists if $\mathbf{A}^T\mathbf{Q}\mathbf{A}$ is positively defined.

For an n-element agent we have:

$$\beta_i(d_i) + \sum_{j \neq i} q_{ij} E[\beta_j(d_j)|d_i] = E(b_i|d_i), \quad (2)$$

where $i, j = 1, 2, ..., n$.

Formalisation of agent decision making process expressed by (2) is a tool necessary for modeling and evaluation of information flow in an autonomous system. Information flow connects agent elements with the external environment described by random variables **X** [4]. For our purposes we define information as knowledge about realisation **X** and the above connection is represented by information structure. This structure is modelled by matrix **C** in which $c_{ij}=1$ if the ith element has obtained (either by observation - sensoring, monitoring - or communication) information about the jth variable **X** realisation (if $c_{ij}=0$ it has not got it). The ith variable **X** realisation can be observed (sensored, monitored) only by the ith element of the agent. It can be informed about other realisations only when communication (information exchange) inside the agent is organised.

The value of information structure defined above is given as:

$$VC=\min E[f(\mathbf{A}, \mathbf{X})|\ \mathbf{C0}]-\min E[f(\mathbf{A}, \mathbf{X})|\mathbf{C}], \quad (3)$$

where $\min E[f(\mathbf{A}, \mathbf{X})|\mathbf{C0}]$ represents the utility of information structure **C0** in which $c_{ij}=0$ for each i and j. Using (2) the VC can be represented by:

$$VC=E[\mathbf{b}^T\beta]. \quad (4)$$

With the mathematical tools presented shortly above it is possible to carry out various simulations of agent functioning in different decision situations [1].

3. NONQUANTITATIVE MODELLING TOOLS

3.1. Decision Tree Classifiers

Decision tree classifiers are used successfully in many diverse areas. Their most important feature is the capability of capturing descriptive decisionmaking knowledge from the supplied data [5]. Decision tree can be generated from training sets. The procedure for such generation based on the set of objects (S), each belonging to one of the classes $C_1, C_2, ..., C_k$ is as follows [6]:

Step 1. If all the objects in S belong to the same class, for example C_i, the decision tree for S consists of a leaf labelled with this class.

Step 2. Otherwise, let T be some test with possible outcomes $O_1, O_2, ..., O_n$. Each object in S has one outcome for T so the test partitions S into subsets $S_1, S_2, ... S_n$ where each object in S_i has outcome O_i for T. T becomes the root of the decision tree and for each outcome O_i we build a subsidiary decision tree by invoking the same procedure recursively on the set S_i.

The above procedure is applied to the training set of objects in Table 1. The training sets are delivered from the analysis based on the quantitative model of agent functioning [1]. Each object is described by the relating attributes and belongs to one of the agent decision classes exchange_information ("yes" in the last column) or do_not_exchange_information ("no" in the last column) [7].

Tab. 1: Training set for agent functioning

external environment	internal environment	type of dynamics	correlation	delay of information	decision
static	independent_actions	0	0	0	no
static	independent_actions	0	0.7	1	no
dynamic	dependent_actions	1.5	-0.5	1	yes
static	dependent_actions	0	0	0	yes
static	independent_actions	0	1	2	no
static	dependent_actions	0	0.5	2	yes
static	dependent_actions	0	-1	3	no
static	independent_actions	0	-1	0	no
static	dependent_actions	0	0.9	1	yes
static	dependent_actions	0	1	1	no

Suppose, for the sake of simplicity, that we are interested in decision making situations involving static environment only. When for this case the set is partitioned by testing on internal_environment and then on correlation, the resulting structure is equivalent to the decision tree shown in Figure 1.

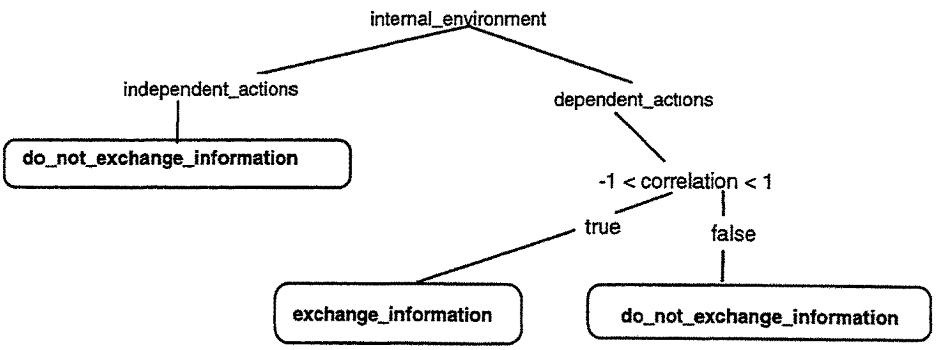


Fig. 1: Decision tree classifier for agent decisionmaking.

The following rules can be delivered from Figure 1

RULE 1
IF an external environment of an agent is static
AND it is described by random variables
AND there is no interaction in the internal environment
THEN communication (exchange of information) between agent elements is not necessary
RULE 2
IF an external environment of an agent is static
AND it is described by random variables
AND there is interaction in the internal environment
AND the relationship between variables describing the external environment is of statistical character
THEN exchange of information between agent elements should be organised

RULE 3

IF an external environment of an agent is static
AND it is described by random variables
AND there is interaction in the internal environment
AND the relationship between variables describing the external environment is given by function dependence
THEN exchange of information between agent elements is not necessary

The simple tree in Figure 1 has been used to retrieve some of the knowledge concerning the functioning of an agent in static environment. The decision tree, once developed, can support decision situations that are not covered by the training set. That is why the production rules can be formulated as a generalised statements. As it has been illustrated the use of decision trees is simple and as effective as the analysis based on a rigorous mathematical model (the production rules formulated above are exactly the same as the ones given in [1]).

3.2. Signed Directed Graphs (SDG) for Modelling and Simulation

Signed directed graphs can be used to build simple qualitative models of complex systems, and to analyse those conclusions attainable based on a minimal amount of information [8,9].

A directed graph, or digraph, is a graph in which all edges are directed [10, 11]. A signed digraph is a digraph with either + or - associated with each edge. SDG nodes are chosen as variables relevant to or representative of the problem that is studied. There is an edge from variable A to variable B if a change in A has a significant direct effect on B. The sign of the edge is + if an increase in A leads to an increase in B, and a decrease in A leads to a decrease in B. The sign is - if the effect is opposite; an increase in A leads to a decrease in B, and a decrease in A leads to an increase in B.

According to the mathematical model evaluation of an information flow depends on the following state parameters: delay of information (d), amount of information (a), dynamics in the external environment (w), variance in the external environment (s), and interaction in the internal environment (q). The above parameters influence the loss in the value of information caused by delay (L1), the loss in the value of information caused by incompleteness (L2), and total loss (LV). SDG will be used to illustrate the process of qualitative reasoning about the informational balance of an agent. Based on relationships and dependencies described by mathematical model, the SDG model as shown in Figure 2 can be developed [12].

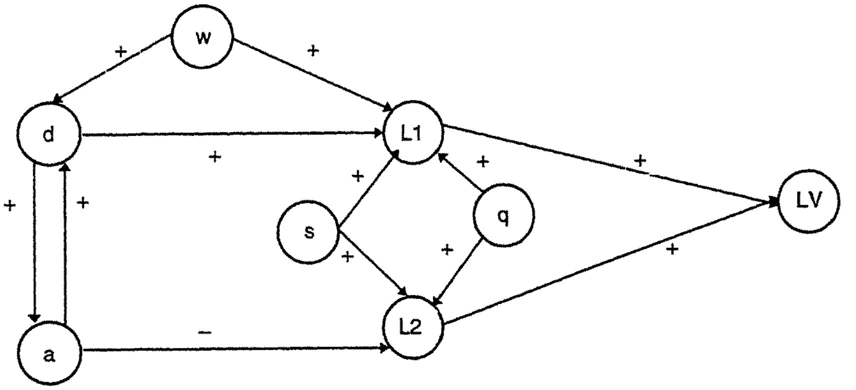


Fig. 2: SDG model of informational balance of an agent

SDG models can be simplified. Two principles are used for the simplification process. The first one is the principle of removal of intermediate nodes and the other one is the simplification of positive feedback loop.

Figure 3 shows the SDG model of an informational balance after the overall simplification.

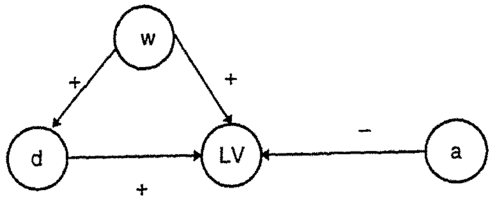


Fig. 3: SDG model of an informational balance - final simplification

The following logic rules can be written for the model in Figure 3.

SDG Rule 1:

IF [d=+] .and. p[dLV]
THEN it is a possible solution pattern for a positive change in d

SDG Rule 2:

IF [a=+] .and. n[aLV]
THEN it is a possible solution pattern for a positive change in a

SDG Rule 3:

IF[w=+] .and. p[wLV]
.and. p[wd]
.and. p[dLV]
THEN it is a possible solution pattern for a positive change in w

Using the above logic rules, the qualitative behaviour of the SDG model can be found. It is easy to notice that the corresponding qualitative states (consistent patterns) for the parameters of our interest are given as follows:

(i) solution pattern for a positive change in d

d	a	w	LV
+	0	0	+

(ii) solution pattern for a positive change in w

d	a	w	LV
+	0	+	+

(iii) solution pattern for a positive change in a

d	a	w	LV
0	+	0	-

The above results of qualitative simulation are exactly the same as quantitative information flow modelling and evaluation (Szczerbicki 1993). For example, they depict the adverse character of two contrary information attributes, i.e. delay and incompleteness. They also show clearly the effects of increasing dynamics in the external environment of an agent. More generally, the results show that as far as the analysis of overall directions of a system behaviour is concerned the simple qualitative model can be sufficient at a minimum level of complexity.

4. CONCLUSION

A substantial part of current research within Artificial Intelligence is focused on developing basic representational paradigms for simplified representation of complex systems. The paper provides the reader with a short review of two such paradigms, namely decision trees and SDG modelling. Its purpose is to understand the role that SDG and decision tree based reasoning may play in developing the flow of information for autonomous multiagent manufacturing systems. Decision trees and SDG modelling are found useful in reasoning about the informational flow evaluation in a manufacturing agent, the important step of multiagent systems development. The approach presented is generally applicable to all possible decision situations faced by single agents and multiagent systems. Further work on possible application of other non-quantitative techniques is needed to fully determine their role in the simplifying process of reasoning about complex issues of information flow evaluation in multiagent systems for which rigorous mathematical analysis is very often too difficult.

5. REFERENCES

1. Szczerbicki, E., "Acquisition of knowledge for autonomous cooperating agent," IEEE Transactions on Systems, Man, and Cybernetics, vol. 23, pp. 1302-1315, 1993.

2. Forbus, K.D., "Qualitative process theory," Artificial Intelligence, vol. 24, pp. 85-168, 1984.

3. de Kleer, J. and Brown, J.S., "Theories of casual ordering," Artificial Intelligence, vol. 29, pp. 33-61, 1986.

4. Szczerbicki, E., "External environment of an autonomous manufacturing agent: dynamics and representation," International Journal of Systems Science, 1996 (in press).

5. Safavian, S. and Landgrebe, D., "A survey of decision tree classifier methodology," IEEE Transactions on Systems, Man, and Cybernetics, vol 21, pp. 660-674, 1991.

6. Quinlan, J.R., "Decision trees and decision making," IEEE Transactions on Systems, Man, and Cybernetics, vol. 20, pp. 339-346, 1990.

7. Szczerbicki, E., "Decision trees and neural networks for reasoning and knowledge acquisition for autonomous agents," International Journal of Systems Science, vol 27, pp. 233-239, 1996.

8. Roberts, F.S., Discrete mathematical models, Englewood Cliffs: Prentice-Hall, 1976.

9. Lau, H.T., Algorithms and graphs, Blue Ridge Summit: TAB Books, 1989.

10. Chartrand G. and Oellermann, O.R., Applied and algorithmic graph theory, New York: McGraw-Hill, 1993.

11. Tinhofer, G., Mayr, E., Noltemeier, H., and Syslo, M., Computational graph theory, New York: Springer-Verlag, 1990.

12. Szczerbicki, E., "Signed directed graphs and reasoning for agents and multiagent systems," International Journal of Systems Science, 1996 (in press).

THE RESEARCH ON INTELLIGENT PLATFORM ISPD FOR INDUSTRIAL SIMULATION PROCESS DRAWING

Y.Y. Jia and Y.N. Leng
Computer Engineering Department
Fushun Petroleum Institute
Fushun P. R. China

ABSTRACT

This paper describes the total structure of ISPD developed with an object-oriented method under WINDOWS and the key technology involved in ISPD. It mainly consists of Screen drawing management subsystem, Industrial components database management subsystem, Intelligent drawing support subsystem, Component database, Drawing files, Process rule and drawing geometry rule database, and application simulation interface template. It involves the following key techniques: intelligent support subsystem, component association, geometric constraint, automatic routing and application simulation interface template. Our aims mainly concentrate on the following aspects: **a.** improving the speed of reference engine to give real-time reaction for drawing operation; **b.** representing or storing the dynamic graph information such as material flowing in the pipeline in graphical file; **c.** simplifying representation of geometric constraint.; **d.** rapid automatic routing strategy; **e.** finding effective interpolation free curves.

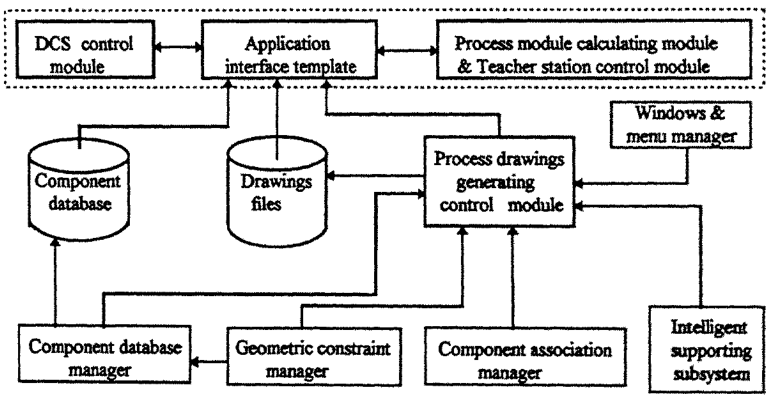
Lastly, we presented an application simulation interface template.

KEYWORDS

Intelligent System , Inference Engine , Simulation , Geometric Constraint , Component Association , Automatic Routing , Free Curve

1. INTRODUCTION

It is often needing to develop simulation training software system one after the other for automatic production equipments that are built a lot recent years in petrochemical industry. These software systems usually involve a lot of process flowcharts , and these flowcharts demand not only graphical information, but also process calculation information; graphical system in simulation software needs not only to handle static graphical information, but also to handle dynamic graphical information such as the material flowing in pipeline, the level of liquid rising up and down and the dynamic displaying of digit etc.. It needs to provide interface in order to



(Figure 1) The system structure outline of platform

transfer needed data among the other modules such as process calculation module, teacher station control module, and DCS module. Therefore, the popular graphical platforms such as AutoCAD does not meet these requirements completely. Based on the above consideration, we have developed ISPD particularly. ISPD is a result of integrating many new CAD techniques such as artificial intelligence, geometric constraint, component association, automatic routing, free curve and so on. The outline of ISPD structure is shown in Fig. 1. Considering that the control structure of simulation graphical interface of various kinds of petrochemical process equipments is similar each to other, we have also developed an application simulation interface template. Using this template, an application simulation software system, oriented to a set of concrete equipment can be easily generated after integrating with process calculation module, teacher control station and DCS control module.

2. INTELLIGENT DRAWING SUPPORT SUBSYSTEM

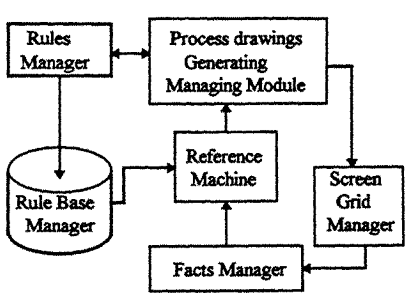
In this subsystem, we use production system to organize geometric rules and process rules that must be followed when drawing the process flowcharts. The knowledge base is a market product. The purpose of this subsystem is to improve the automatic drawing level, and its structure outline is shown in Fig. 2. Usually, control strategy is a deterministic factor in production system. The control strategy in our intelligent subsystem and the principles that this subsystem works on will be roughly depicted. The facts and rules for petrochemical process flowchart drawing both can be represented in normal production form:

IF { predicate { ∧ predicate { ∧ ......}}} THEN { predicate { ∧ predicate...... }}.

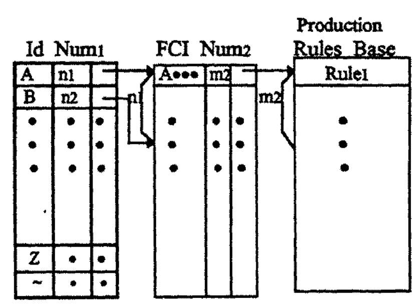
For instance, IF { HORLINE(x) ∧ VERLINE(y) ∧ CROSS(x,y) ∧ POSITION(y,z)} THEN CUT(y,z) represents the rule often used in process flowchart drawing, it means that "if horizontal pipeline crosses with the vertical one, then, horizontal line will break vertical line", i.e. the rule "vertical line is broken ,but horizontal one is not ", for brief.

Generally, in petrochemical process flowchart drawing, the basic element that formed a fact or a rule is a predicate, not a clause. Moreover, the number of type of facts is few, the type of facts is simple and unchanged. Therefore, in order to ease the matching and searching process, we employ the first character index concept. To form a first character index of a fact or a rule, the predicates that form a fact or a rule conditions are sorted alphabetically first, then ,the first character of each predicates is taken out to concatenate a first character string, that is the first character index needed. the first character index of the example rule just described above is CHPV, not HVCP.

To use the first character index technique as a predicate matching method in order to reduce the backtracking in reasoning process, we organizes the knowledge base as shown in Figure 3. In knowledge base, rules are alphabetically sorted and arranged in 26 classes. The matching speed-up strategy are as follows:



(Figure 2) The structure outline of intelligent supporting subsystem



(Figure 3) The structure of rules base with 2nd level first character index

(1) The effective fact generation mechanism based on grid management

There are very much information involved in one drawing. If all of them are transformed into facts, the global database certainly will become very larger and larger, and the reasoning will become difficult and slow. Obviously, transforming all information into facts is neither adaptable, nor necessary. When the user creates, moves, deletes and copies the graph primitive, only the information in a few grids that the graph primitive occupied will be changed and only these changed information need to be formed into facts for activating reference to reason, A lot of unchanged information can be left unprocessed, the facts generated in this way are called effective facts. Obviously, this management can reduce the number of facts in global database dramatically.

(2)The first character pre-matching mechanism.

The first character string of a generated fact matches the first character string index of rules in sequence, if the matching is successful, then the unification algorithm will be called to complete the reasoning, if it is not, then matching goes to next rule. In this way, the number of calling unification algorithm will be reduced, and it implies that the backtracking times will be reduced, then, the reasoning will be speeded up.

3. COMPONENT ASSOCIATION

This function is based on the "layer" group operation concept of AutoCAD package. That is, the operation such as move, copy and etc. can be applied to associated components as if they are integrated together. In order to obtain the ability to display material flow in pipeline dynamically, a data structure called flow path association list (FPAL, for short) is needed, and the equipment components in the flow path are all stored in this list according to the flow sequence. The structure of the list is shown in Fig. 4.

The connection relationships among the components can be devided into serial and parallel cases, Here the Pointer$_i$ points to the ith association item in FPAL. Pointer$_i$ points the address of the ith component in the serial case, or points the address of parallel connection index list in the parallel connection case.

Parallel connection means that the material will go to several components simultaneously from the ith components, in this case, pointer$_{i+1}$ does not point to one component, but points to a parallel connection index list(PCIL for short). All parallel connected components are recorded in this index list, the pointer PP$_j$ points to the Pointer jth components in the list. This platform provides operations of creating, modifying, deleting and dynamically displaying for the two lists.

FPAL Identifier	the name of material	the parameter of process calculation	the number of components	association item1	••••••	association item2	association item$_n$

(a) Flow path association list

PCIL Identifier	the parameter of process calculation	the number of component	parallel pointer1	parallel pointer2	••••••	parallel pointer$_m$

(b) Parallel connection index list

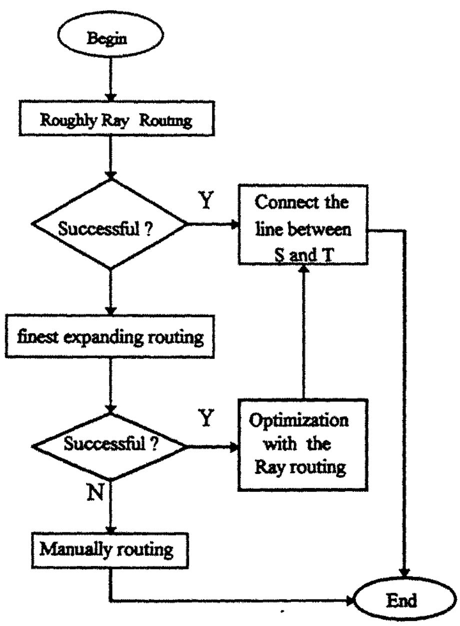
(Figure 4) The data structure used in component association

4. GEOMETRY CONSTRAINT

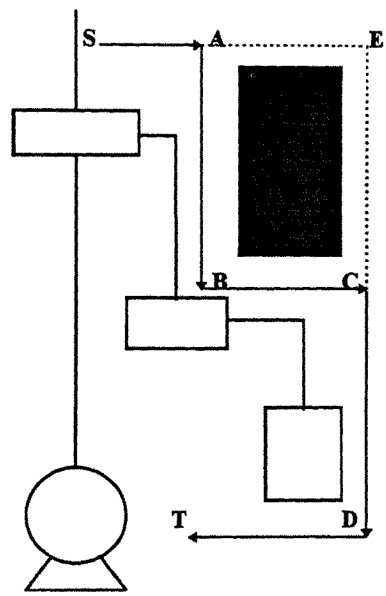
The geometric constraints among graphical elements are complex in coupling. It will form a constraint network if the relationship might be described naturally. In this network, each node represents a graphical elements, directed arch between nodes means that a geometric constraint exists between them. the weight on the arch is the type of that constraint. Article [1] provides a constraint dynamic propagation algorithm based on this kind of constraint network. Article [2] uses the structure called reduced tree to represent the constraints between graphical element nodes and dramatically simplifying the representation of constraint. The constraints in process flowchart simulation mostly appear between semi-ellipse and straight line, and the constraint types are simple enough. The geometrical constraints needed to be considered in this platform are only limited in the following cases: line segment tangenting to a semi-ellipse on the end of line segment; symmetry between two seme-ellipess; parallel of two line segments and perpendicalarity of two line segments. Considering the concrete requirements stated above, the original bilateral coupling geometric constraints are decomposed into one directed constraint. For instance, the expression " Master GCP $\xrightarrow{GCT}$ Slave GCP " is an one directed geometric constraint, here GCP is a geometric contraint primitive, Gct is the constraint type in this expression, the two GCPs are not in the same priority level, the dynamic propagation occurs only when the parameters of master GCP has been changed; the parameter changes of slave GCP will not affect the master GCP. thus, geometric constraint propagation becomes serial one in linear list. The propagation algorithm becomes very simple. The detailled algorithm is omitted here due to the volume limitation.

5. AUTOMATIC ROUTING

The work amount of pipeline drawing takes 50-70 percent of process flowchart drawings. Therefore it is necessary to introduce the automatic routing technique of ICCAD to this platform. We studied various automatic routing algorithms deeply, and found that the Maze routing algorithm has the advantages of strong detour ability, good adaptability and good flexibility. but this algorithm also has the disadvantage of slowness due to expanding pixel one by one; the finest search routing algorithm has overcame the blind search disadvantage of maze algorithm, it first expands the neighbor pixel that leads to the goal, thus the speed has been improved. But, this algorithm can not overcome as the same disadvantage that the corner point are too many as in the maze algorithm. Ray routing algorithm completes routing by radiating ray to all directions. Therefore, the routing speed is improved greatly. But, the detour ability of this algorithm is very poor. When the obstacles are complex, it can not route through two points successfully when using this algorithm. Thus, it is a incomplete algorithm for routing. Putting the advantages of several algorithms together, we employ a synthesizing algorithm described as follows.



(Figure 5) The outline of synthesizing routing strategy



(Figure 6) The optimization by ray routing

(1) roughly routing by using ray algorithm, if success then stop; otherwise goes (2)
(2) routing in detail by using finest search routing algorithm. if success goes (4)
(3) interactive routing manually.

(4) optimizing the routing result to decrease the number of corner points by using ray routing algorithm.

The principle of this algorithm is shown in Fig. 5. The optimizing process in this algorithm is briefly described as follows: choosing two corner points that are not on the same line segment first, then radiating rays from these two points respectively, if two rays meets each other and there are no obstacles in the path of rays, the original path between the two corner points can be replaced by the new path of ray. Thus the number of corner points is reduced. For example, the primitive routing between points S and T is SABCDT, there are total 4 corner points in the routing line. After optimization, the routing path becomes SEDT, There are only two corner points E and D in the routing (see Fig. 6).

6. FREE CURVE

we have studied various interpolated free curves. Of that interpolated curves, the commonly used 3-spline uses exponent function 1, x, x^2 and x^3 as its basis function; by using variable parameter simultaneously,its formula is a piecewise polynomial $P(x) = a+bx+cx^2+dx^3$. When using this method, it is necessary to solve a linear equation system that is formed from the given point values, 1st derivative and 2nd derivatives to get coeffiencents a,b,c and d. The typical parameterization method has the following drawbacks: a. needing to solve the linear equation system; b: needing to provide two additional constraint conditions at the end points; c. local changes in the curve will propagate over the entire curve, so that the error correction may cause undesired effects elsewhere. However, the multi-node spline interpolation curve uses a group of uniform B-spline functions as its basis functions. Its interpolation formula is formed with typical interpolation formula construction method. B-spline basis function have good mathematic properties and local deformability, therefore, its representation is very easy to be obtained, not needing to solve the linear equation system as doing in 3-splines. As to the detail deduction, see reference [3]. Generally speaking, the multi-node interpolation spline curve has some good properties as follows:

(1) The interpolation speed of this curve is much faster than the interpolation speed of 3-spline;
(2) Local deformability just as B-spline curve possess;
(3) Easily Computing, neither needing to solve the linear equation system as doing in the case of 3-splines, nor needing to provide the additional constraint conditions at the end points.
(4) Simple analysis expression just like Bezier curve.

For instance, 2nd uniform B-spline basis functions are $B_{0,2}=0.5(t-1)^2$; $B_{1,2}=0.5(-2t^2+2t+1)$; $B_{2,2}=0.5t^2$, then, the expression of 2nd multi-node interpolation spline curve is using 2nd uniform B-spline basis functions is

$$CS^2(t) = \begin{cases} [1 \;\; S \;\; S^2]\, B^2\, [Pj\text{-}1 \;\; Pj \;\; Pj\text{+}1 \;\; Pj\text{+}2]^T & \text{in the case that } S \in [0, 0.5] \\ [1 \;\; 1\text{-}S \;\; (1\text{-}S)^2]\, B^2\, [Pj\text{+}2 \;\; Pj\text{+}1 \;\; Pj \;\; Pj\text{-}1]^T & \text{in the case that } S \in [0.5, 1] \end{cases}$$

$$\text{where, } B^2 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0.5 & 0 & 0.5 & 0 \\ 0.75 & -1.75 & 1.25 & -0.25 \end{bmatrix}$$

7. APPLICATION SIMULATION INTERFACE TEMPLATE

In order to provide flexibility for different kinds of users, we have developed application simulation interface templates under DOS and WINDOWS respectively. By using these templates, users can generate their own application simulation interface conveniently. Application result shows that the template is very applicable and effective.

8. CONCLUSION

This platform is developed in BORLAND C++ under WINDOWS, the program paradigm is OOP. It is not only a petrochemical process simulation oriented intelligent platform, but also a graphical software platform satisfying requirements of multifuction and various projects.

9. ACKNOWLEDGMENT

We are thanks greatly our students J.M. Wei, F.S. Zhang, X.N. Meng, Y.Y. Zhang, Z.H. Xie,and Z.Y. Chen for their effective works in the realization of this platform.

10 REFERENCES

1. Suzuki H, And. H, Kimura F, Geometric Constraints and Reasoning for CAD system Computer and Graphics Vol.14, No.2 212~224 1990
2. Luo H, Chen L.P, and Zhou J. 'Principle of Geometric Modelling Based on Constraint and Its Realization' Application Reaserch of Computers Vol. 13, No. 5 16~19 1996.
3. Qi D. X., Liang Z. S. and Ma S. L. Graphics and Its Exploration in Computer The Press of Jilin University 38~48 1989
4. Wei J. M. and Jia J. Y. A Graphical Software for Industrial Process Drawings The Journal of Fushun Petroleum Institute Vol. 12, No.4 113~118 1995
5. J. Encarnacao Computer Aided Design Modeling, System Engineering, CAD-Systems Springer-Verlag Berlin Heidelberg New York 142~158 1980

RESEARCH ON A FUZZY CONTROL METHOD BASED ON EXPERT SYSTEM FOR THE CAVITY OF SALINE PRODUCTION

Liudong, Yin Guofu, Chen Jiuqiang and Hu Xiaobing
Department Of Mechanical Engineering
Sichuan Union University
Chendu 610065, China

ABSTRACT

This paper deals with a "quasi" closed loop control method incorporating fuzzy logic based on expert system for invisible cavity of saline production in Zigong saltworks, China. The system structure of an unique adaptive controller is prensented for the brine concentration and varibale volume of cavity, which are able to compensate by the water flow-down control. The fuzzy control strategy in the system is provided by the expert system. The using results in the saltworks shows that the algorithm of self adaptive control makes the brine concentration stable and the cavity shape control efficiently.

KEYWORDS

Expert System, Fuzzy Control, Cavity Shape of Saline, Self Adaptive Control

1. INTRODUCTION

In saline production, the cavity shape plays a role in the life, the produtivity and the safety of the salt well. The shape is utmost important to the production and the cost in the case of two wells connected. But because the cavity lies in underground about one thousand meters away, the shape is difficult to monitor. The developement of the cavity is affected by many factors such as the complicted geological structure and slow variation in rate. So it is difficult to give a mathematical model. The cavity shape can't be effectively controlled by common algorithm such as PID regulator. In the past process of saline prodution, the cavity was controlled completely by the operators' experience, which still makes the shape hard to define in the end. There wasn't any good way to solve this problem for a long time. This paper introduces a principle and algorithm of fuzzy control based on expert system for the shape which having been simulated by visualization method on the computer screen.

2. INTELLIGENT CONTROL STRATEGY

2.1 Description For Developing Process Of Cavity

Saline production is a cyclic process of water-in and brine-out by injecting high pressure water into cavity to dissolve the underground salt bed, which makes the cavity expand continually. In order to let the cavity be closed to "arch" shape, the developing process is divided into several phases. In the same phase, the top salt layer can be protected by oil while the side bed is being dissolved. According to theoretical analysis and experiment, the developing process of cavity can be described as follows:

$$\mathrm{KQ}\frac{\partial \mathrm{C}}{\partial \mathrm{Z}} - \mathrm{KD}\,\pi r^2 \frac{\partial^2 \mathrm{C}}{\partial \mathrm{Z}^2} - 2\mathrm{KD}\,\pi r\,\mathrm{tg}\,\theta \frac{\partial \mathrm{C}}{\partial \mathrm{Z}} - \frac{2\pi r}{Cos\theta} + \mathrm{K}\,\pi r^2 \frac{\partial r}{\partial \mathrm{t}} = 0 \tag{1}$$

$$\frac{dr}{dt} = \frac{\mathrm{W}}{\rho Cos\theta} \tag{2}$$

where:

K : Moore quality	Z : height of cavity
Q : flow of water	W: dissolving rate
C : concentration of salt brine	ρ : density of salt bed

D : diffusivity θ : side angle of salt bed

r : radius of cavity per unit bed t : time

These differential coupled equations need additional empirical formulas to get answer. The developement of cavity possessing of nonlinear characteristic with slowly time delay, this mathematical model can't be changed into the model for control in application.

2.2 Intelligent Visualized Control System

The accurate mathematical model for cavity control is difficut to build up. But the conclusion for control is able to be obtained from theoretical analysis and experiment: the cavity shape is defined by r and θ , and the key factors which have influence upon them are Q and C in a certain stage. Simultaneousely, C and Q are most important to the productivity and the safety of the salt well. Fig.1 gives a scheme of visualized intelligent control method for the cavity.

The water flow Q is controlled by the fuzzy logic controller (FLC) in producing process (see Fig.1).The visualization system (VC) displays the cavity shape on the computer screen according to Q, C and other parameters in a certain stage. Also VC inputs all of these parameters to the strategic expert system (SES). The SES makes the decision on a suitable value C_d of concentration and modifies the fuzzy control strategy at the present process.

Originally, the system of cavity shape control only possessing of open loop plant, which is only based on water input and brine output, so the shape can't be known actually. After VS is added, the shape is able to be shown and modified by people or by computer automaticlly. This method makes the open loop becomes closed. Because the visualized shape, which may be not true, only gives an idea of experience, this system is called "quasi" closed loop.

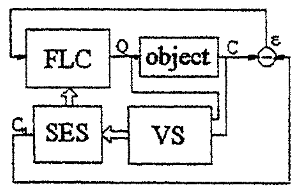


Fig.1: Intellgent visification control system

2.3 Intelligent Decision

Intelligent strategy for control, which includes the technologic parameters at present and the strategic knowledge, is decided by SES fuzzy inference. Applying the theorey of fuzzy set is able to simplify the process of inference and has a reduced rules in the knowdelge base.

The technological parameters correspond to the developing process of cavity. Making the strategic decision must refer to them. Supposing P_i represents the set of the technological parameters in No. i process, the decision can be made as follows:

$$\text{IF} \quad W_i^1 \,\&\, W_i^2 \,\&\, W_i^3 \,\&\, W_i^4 \,\&\, W_i^5 \quad \text{THEN} \quad P_i$$

...

where W_i^j represents:

j=1: phases . j=2: point of ejecting water j=3: point of extracting brine

j=4: cyclic mode j=5: productivity

SES makes the fuzzy decision according to the technologic parameters and the two-dimension size $R(r, \theta)$ of the cavity obtained from VS. Considering respectively fuzzy linguistic value of $\dot{r}$ and $\dot{\theta}$ as: {PB (positive big), PM (positive middle), PS (positive small), ZE (zero), NS (negative small), NM (negative middle), NB (negative big)}, the rules for decision can be expressed as follows:

case $\mathbf{P}_1$:

IF $\dot{r}_1$ is A_1 and $\dot{\theta}_1$ is B_1 THEN C_i^1

IF $\dot{r}_2$ is A_2 and $\dot{\theta}_2$ is B_2 THEN C_i^2

...

IF $\dot{r}_n$ is A_n and $\dot{\theta}_n$ is B_n THEN C_i^n

...

where $(A_1, A_2, \cdots, A_n)$, $(B_1, B_2, \cdots, B_n)$ and $(C_i^1, C_i^2 \cdots, C_i^n)$ are fuzzy subsets of $\dot{r}$, $\dot{\theta}$ and C_d.

The fuzzy control strategy above is able to be described as: steady development of the cavity is defined by r and θ when the technologic parameters are acnowkedged, furthermore:

(1) if r increases rapidly and θ is bigger, dissolving rate W is too big. The strategy of improving concentration is adapted.

(2) if r increases slowly and θ is smaller, dissolving rate W is too small. The strategy of decreasing concentration is adapted.

...

2.4 Strategy Of Fuzzy Control for Concentration

Concentration control has relation to the shape of solution cavity. Steady development of the cavity repuires the stablity in concentration, which means the dissolving process is steady. Vaiation in concentration also possesses of a nonlinear characteristic with slow variation in time delay, which is difficult to describe in a accurate mathematical model. The method of fuzzy control summarizes the people's experience into the rules which can be accepted by controller. It is based on fuzzy set theory that quantifying the variable with linguistic variable. Fuzzy control is similar to that of the operator's experience, so it's fit for the nonliner object control.

Once the value of C_d obtained from SES, the fuzzy controller stabilizes the concentration beyond its tolerance.The controller input are error E and change in error Ec, and the ouput is increment of water flow △Q. The rules for control can be summarized as:

IF **E** is G_k and **Ec** is H_k THEN △**Q** is U_k

...

where G_k, H_k and U_k respectively represents fuzzy subset of E, Ec and △Q under the rule K.

This controller structure is able to effectively control the concentration in a certain volume of the cavity. The dissolving area extends along with the volume increasing, and flow of water ought to increase although variation in concentration is same as ever. If only this structure is used, the flow may be regulated frequently, futhermore, it will cause the concentration overshoot. For this reason, the self adaptive quality should be added in the controller in order to meet the needs for control when the volume changed.

3 FUZZY ADAPTIVE CONTROL

3.1 Improvement On Structure Of Fuzzy Controller

Because it is too complex and difficult to fuzzifying the volume of cavity, an alternative to the method for control is to regulate the water flow according to the trend of variation in concentration. When the volume becoming bigger, the controller ouput a proportional factor of flow to stabilize the concentration. The improved structure for fuzzy controller see Fig.2.

The improved fuzzy controller has two main components: a fuzzy controller for flow and a adaptive regulator for proportional factor. The fuzzy controller consists of knowldge base, fuzzification, reference and defuzzification. The controller input are still error E and change in error Ec, but the ouput is quantified value u of the increment △Q. When the volume of cavity changes, the adaptive regulator compensates the water flow by magnifying k_Q. So, the total flow of water is:

$$Q = u \times k_Q + Q' \qquad (3)$$

where Q' is the value of water flow last time.

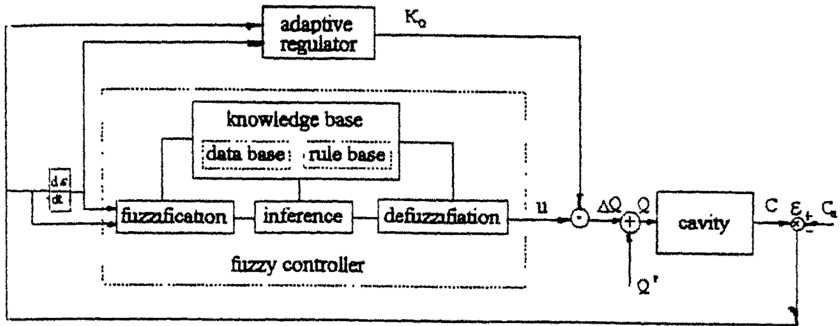


Fig.2: The improved structure for fuzzy controller

3.2 Fuzzy Logic Control

The fuzzification of error E is classified into seven grades with {PB, PM, PS, ZE, NS, NM, NB} in the area of fuzzy set, for the change in error Ec with {PB, PS, ZE, NS, NB}, and for the output variable Δ Q with {PB, PM, PS, ZE, NS, NM, NB}.The shapes of membership function of E and Ec are typtically three triangular, and the bar shapes are used in membership functions of Δ Q. The membership functions of each variable are showed in Fig.3(a), Fig.3(b) and Fig.3(c).

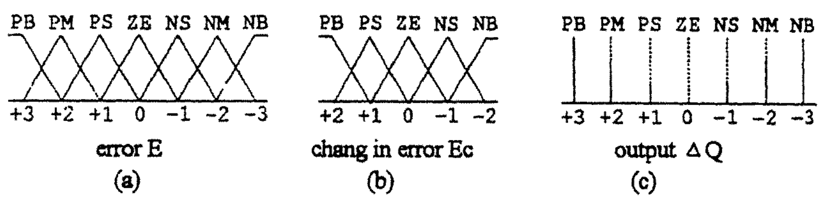


Fig.3: Membership functions of each variable

The rules for control in application are listed in Table 1.

Ec \ u \ E	NB	NM	NS	ZE	PS	PM	PB
NB	PB	PB	PM	PS	NS	NM	NB
NS	PB	PM	PS	ZE	NS	NM	NB
ZE	PB	PM	PS	ZE	NS	NM	NB
PS	PB	PM	PS	ZE	NS	NM	NB
PB	PB	PM	PS	ZS	NM	NB	NB

Table 1: Rules for control

3.3 Algorithm Of Adaptive Adjustment For Proportional Factor

Consider the input-output function of controlled object as follow:

$$Q = g(C, \dot{C}) \tag{4}$$

or

$$\dot{C} = f(C, Q) \tag{5}$$

Noting the equation 5, the differential equation is:

$$\delta\dot{C} = \frac{\partial f}{\partial Q}\delta Q \tag{6}$$

If the micro-variable Δ Q inputs the cavity at K, after τ interval, the output of concentration is:

$$\Delta C(k+\tau) = \tau T \delta \dot{C}_k$$

$$= \tau T \frac{\partial f}{\partial Q} \delta Q_k \quad (7)$$

where τ is time delay, T is sampling interval.

Because the input Q(k) and the output C(k+ τ) in this system are all known, and the cavity volume can be approximatively regarded as a constant in two adjacent time. The partial differentiating equation of the function can be expressed by :

$$\frac{\partial f}{\partial Q} \approx \frac{C_{k+\tau} - C_k}{Q_k - Q_{k-\tau}} \quad (8)$$

So the modified value of flow at K is:

$$\Delta Q = P^{-1} \Delta C(k+\tau) \quad (9)$$

$$P^{-1} = (\tau T \frac{\partial f}{\partial Q})^{-1}$$

The modified value of k_Q is:

$$k_Q = \frac{\Delta Q}{u} + k'_Q \quad (10)$$

The water flow at $(k+\tau)$ is:

$$Q = k_Q \times u + \sum_{i=1}^{k} Q_i \quad (11)$$

4 APPLICATION

The method proposed in this paper was applied in Zigong saltworks, China. Brine concentration was steadly controlled on the whole, and overshooting was advoided. According to the statistical data till 1995, the safe phase was prolonged for another 1.28 times than the other well arrays, and the production of salt was improved 38.96%. The results shows that the cavity shape is developing in "arch" shape, which proves that this method is effective in engineering application.

5 CONCLUSION

The method proposed in this paper solved the problem of control for the invisible shape of saline cavity expanding under the ground. Simulation by visualization method on the computer screem creats a "quasi" closed loop system instead of the open loop employed ever. The algorithm of self adaptive simplifies the model for control while the cavity volume variating, which makes the brine concentration stable and the cavity be controlled to the required shape. The application in the saltworks proves that this method can be satisfied for saline production.

6 REFERENCES

1. Yuanxong Lin, Salt Solution Mining Technique, Sichuan Publishing House of People,1990.12
2. A.Ismael B.Hussien R.W.Mclaren .Fuzzy Neural Network Implementation of Self Tuning PID Control systems.IEEE Inter.Symposium. Columbus,Ohio,1994.
3. James G.Dawson,Zhangqiang Gao.Fuzzy Logic Control of Linear Systems with Variable Time delay.IEEE Inter.Symposium, Columbus,Ohio,1994.
4. C.C.Lee, "Fuzzy Logic in Control Systems:Fuzzy logic controller--Parts 1&2 " IEEE Trans. on Sys. Man, and Cybermetics, Vol 20, No.2, pp404-435 March/April 1990.

THE ROLE OF KNOWLEDGE RICH MACHINE CONTROLLERS WITHIN FLEXIBLE MANUFACTURING

N.N. Gindy, N.J. Pickett and S. Owen
Department of Manufacturing Engineering & Operations Management,
University of Nottingham, University Park,
Nottingham, England.

ABSTRACT
This paper introduces ongoing research aimed at developing a manufacturing cell control strategy to exploit the capabilities of 'Knowledge Rich Machines Controllers' (KRMCs) forming a new collaborative strategy for rapid component manufacture. A detailed explanation of the structure of a KRMC describes a knowledge-based approach to geometric and technological reasoning for the purposes of: feature refinement; process, tool and fixture selection; and set-up determination, based upon a machine tool's capability. An architecture is presented demonstrating the integration of feature-based CAD, CAPP and Production Planning for the communication and distribution of product data to each machine for processing. Finally, an example of the application of a high level cell controller for the management of several KRMCs is illustrated.

KEYWORDS
Collaborative Strategy, Knowledge-Base, Geometric and Technological Reasoning.

1. INTRODUCTION
The major challenges currently facing the manufacturing industry can be characterised by increasing demands for shorter lead times, more flexibility, improved quality in production, and most importantly faster transition from design to manufacture. These challenges are being addressed through a combination of technological developments such as: feature-based design; non-linear process planning, intelligent machine tools [1]; and open systems architectures, etc.

In examining these state-of-the-art developments more carefully, several key technologies must be taken advantage of before such improved methodologies and concepts are able to make their promised impact on the daily operations of manufacturing companies. Exploiting technologies such as: local machine tool controllers to optimise the performance of the manufacturing tasks; enhancements to feature technology in NC part programming to increase the level of automation; and improved geometric reasoning from CAD and CAPP, will have a significant impact on the flexibility and responsiveness of manufacturing systems.

Manufacturing research needs to address the above problems and explore the capabilities of local machine controllers to bridge the gap between the 'promised land' and the current manufacturing reality. The aim of this research is to investigate the capability boundaries of the machine tool controller to allow more effective delegation of process and production planning activities within a cellular manufacturing environment.

2. KNOWLEDGE RICH MACHINE CONTROLLER
Today's modern machine tool control systems provide extensive capabilities [2], such as machine diagnosis, NC programming, gauging, thermal compensation etc. To extend these capabilities and to introduce production and operations planning at the machine tool controller level,

an integrated architecture for CAD, CAPP and Production Planning has been formulated in which a 'Knowledge Rich Machine Controller' (KRMC) forms an integral part.

A KRMC is a self-contained virtual device, with local intelligence capable of describing the status, capabilities and limitations of an associated machine tool (host). The function of the KRMC is to utilise the machine tool specific information and determine through a combination of geometric and technological reasoning the machine tool's contribution to a component's manufacture with minimum human intervention. This can be considered as a combination of the host machine tool's capability (e.g. anatomy of the machine tool and the operations available) and the resources (e.g. cutting tools, fixtures, retrofits, etc.) available. Using the local knowledge, a KRMC is able to communicate with other integrated applications (e.g. CAD, CAPP, etc.) in order to extract the necessary information to proceed with the component processing.

The structure of a KRMC is shown in Figure 1 and consists of:

- a Geometric and Technological Reasoner (GTR) to determine the intermediate and low-level operations;
- a dispatcher to monitor the work in progress and to manage the queue of jobs waiting to be processed;
- an NC-Code generator and an optional inspection code generator;
- a machine tool controller system to drive and control the host machine tool;
- local knowledge-bases to store and maintain machine tool, process and resource information.

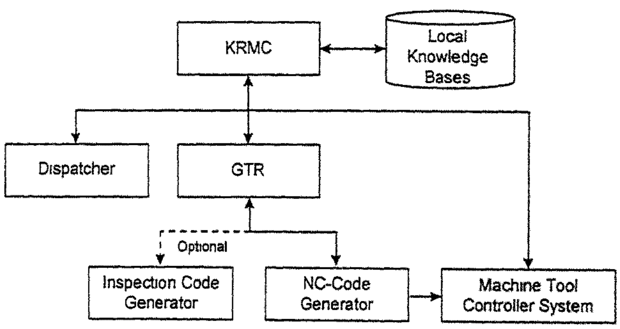


Fig. 1 : Knowledge Rich Machine Controller Structure

Each KRMC is able to initially determine its host's contribution to the component's manufacture by simply matching the operations listed in a machine-independent process plan for the component with those provided by the host. The KRMC can then further interrogate and process the feature based component model using its machine specific geometric and technological reasoner to identify and optimise set-ups, refine features, etc.

In order to achieve a realistic representation of a machine tool for the purpose of geometric and technological reasoning, a large amount of data has to be stored. This includes describing the basic characteristics of the host machine tool (e.g. table size, spindle speeds, etc.). Although supplementary data such as machine location, age and last service, may not directly affect the result of the geometric and technological reasoning, they can influence the selection of a machine tool for the purpose of production planning.

2.1 Geometric and Technological Reasoning

The heart of the KRMC is a machine specific geometric and technological reasoner (GTR). Its principle role is to interrogate and process the feature-based component model and the technological information produced during the high level machine independent process planning. The geometric and technological issues relevant to the host machine capability can then be identified. The functionality of the GTR, as its name suggests, is provided by two separate modules, namely the geometric reasoner and the technological reasoner.

The task of the geometric reasoner is to process the geometric information of the feature-based component model after the design stage, in order to:

- identify external access directions and potential access directions [3] for cutting tool access;
- identify parent-child feature relationships for feature sequencing.

The technological reasoner's task is to process both the geometric and technological component information using the capabilities of the host machine in order to:

- split and merge features based on the resources available;
- determine the number of component set-ups;
- select the required cutting tools and conditions;
- create feature sequences, based on parent-child and tolerance relationships;
- select the required fixtures.

3. INTEGRATED ARCHITECTURE

The integrated architecture for flexible manufacturing incorporating KRMCs is split into two levels, the factory level and the KRM level, as shown in Figure 2.

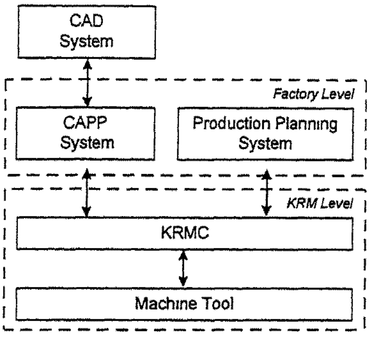


Fig. 2 : Integrated Architecture

The factory level contains:

- a high level CAPP system to create high level machine-independent process plans, detailing the required sequence of operations and set-ups for each feature based on Resource Element (RE) [3] allocation (where an RE is a set of machine independent operations that can be obtained from one or more machine tools);
- a production planning system to process, co-ordinate and control the contribution of each KRM and the resources within the factory, whilst ensuring the desired components are manufactured at the right time, in the correct quantities and with optimised performance.

The KRM level contains:

- a KRMC to process both the feature-based component model and the high level machine-independent process plans generated at the factory level;
- the appropriate machine tool associated with the KRM.

3.1 Production Planning using Multi-Level Machine Operation Planning

Traditional process planning [4] usually consists of two levels, a macro and a micro level (high and low). The initial set-up plan, machine selection, process and raw material selection are done at the macro level. The remaining operation step and cutting tool selection are done at the micro level. Due to the approach taken to process planning using KRMCs in this project, it has been necessary to introduce a third level in between macro and micro, termed the 'intermediate' level, shown in Figure 3 below This level is a machine tool-specific level for utilisation with KRMCs only.

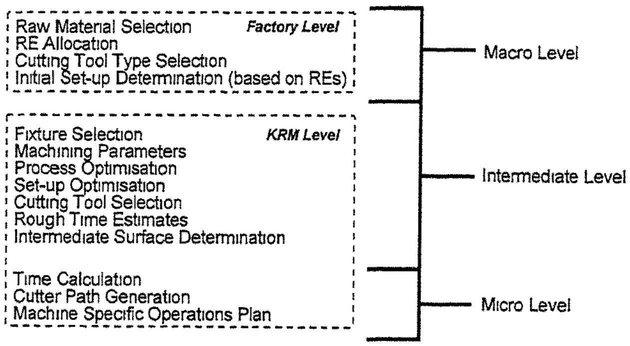


Fig. 3 : Multi-Level Operations Planning

When a feature-based component model is entered into the system, the high level CAPP system generates high level non-machine specific process plans or 'macro' plans. These contain the required set of REs, together with a list of component set-ups. The CAPP system then distributes the machine independent plans to all of the appropriate KRMs containing the required REs.

When a KRMC receives the plan, it determines its host machine contribution (whether repeated in the factory or exclusive to that machine) towards the component's manufacture by enriching the level of information conveyed in the plan by the application of machine tool specific knowledge, which includes: cutting tool selection, machining parameters, rough time estimates, etc. The machine tool specific plan or 'intermediate' plan is then sent to the production planning system to form the KRM's *bid* for the work. The intermediate level of operation planning allows the utilisation of rough time estimates in the initial optimisation of production planning at the machine shop level.

When the production planning system has received all *bids* from all KRMs, it compares their contents (rough time estimates, required resources, etc.) together with the production schedule for each bidding KRM to determine which machines are the most appropriate for the component's manufacture. The production planning is then able to form an optimised schedule based on the time estimates given by the selected KRMs and hence forms a collaborative manufacturing cell of KRMs whose combined capability is sufficient to completely manufacture the component [5].

As each KRM completes its contribution to the component's manufacture, it reports the actual time taken to the production planning system. The production planning system is then able update the component schedule and identify any problems.

3.2 Example Application

Consider a feature-based component with 3 features. When the component enters the system, high level process plans (based on the component requirements) are generated and distributed to the appropriate KRMs. Upon receiving the feature-based component model, each KRMC determines its host machine tool contribution to the component's manufacture, as shown in Figure 4 below

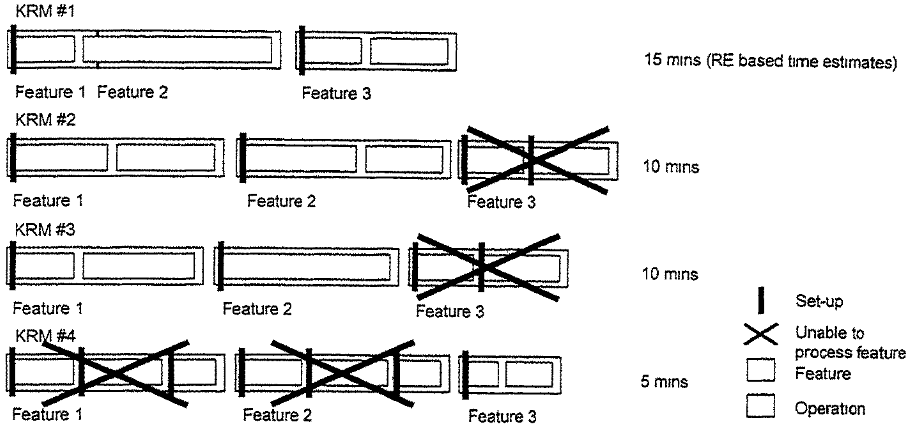


Fig. 4 : Production Plan

In this example, KRM #1 concludes that it is able to merge features 1 and 2 into a single feature, which requires a single set-up and 2 operations. It also determines that it can do feature 3 in a separate set-up with 2 operations. This information, together with the rough time estimates for each RE, is passed onto the production planning system to form the bid for KRM #1 at the component level.

Similarly KRM #2 and #3 both conclude that they are able to do features 1 and 2, using two separate set-ups, but are unable to operate on feature 3, as the operation or cutting tool required is not within its host machine tool's capability. Again, this information together with the rough time estimates for each RE is passed onto the production planning system to form their bid.

Finally, KRM #4 concludes that it is able to manufacture feature 3 in a single set-up, but is unable to operate on features 1 or 2. This information together with the rough time estimates for each RE is passed onto the production planning system.

It can be seen from the above example that there exists some repeated capability amongst the KRMs, although the repeated capability may not be achieved the same way and that each solution or bid is unique to the KRM which produced it. It is the task of the production planing system to consider each bid and select the most appropriate KRMs, as previously described and hence establish a manufacturing cell and route for the component's manufacture. However, other factors such as machine tool location, intermediate component transport and problems with NC-Code generation must be accounted for, as they may cause additional problems to the chosen route and schedule.

4. SUMMARY AND CONCLUSION

This paper discussed research aimed at developing a new collaborative manufacturing strategy, which exploits the capabilities of KRMCs to achieve rapid component manufacture. A detailed explanation of the structure of a KRMC has been given and an integrated architecture for CAD, CAPP and Production Planning through KRMCs has been presented.

The research presented in this paper is just the first step towards the complete development of a collaborative manufacturing strategy, utilising KRMCs. The investigation so far has suggested that exploiting the capability of a local machine tool controller can provide a suitable means for machine level operations planning and also provides a platform for the integration of CAPP and Production Planning.

5. ACKNOWLEDGEMENT

This work is supported by EPSRC Grant GR/K38472. Their financial support is gratefully acknowledged.

6. REFERENCES

1. Wright, P.K. and Bourne, D., Manufacturing Intelligence, Chapter 2, Addison-Wesley Publishing, New York, 1988.
2. Nagurka, M.L. and Englert, P.J., 'Toward an Intelligent Machine Tool for Flexible Manufacturing', Robotics & Computer-Integrated Manufacturing, Vol. 6, No. 3, pp. 229-236, 1989.
3. Gindy, N.N., Huang, X. and Ratchev, T., 'Feature-based component model for computer-aided process planning systems', Int. J. Computer Integrated Manufacturing, Vol. 6, Nos. 1 & 2, pp 20-26, 1993.
4. Chang, T., and Wysk, R., An Introduction to Automated Process Planning Systems, Chapter 4, Prentice-Hall, New Jersey, 1985.
5. Gindy, N.N. and Saad S.M., 'Resource-Based Scheduling in Virtual Manufacturing Environments', Proceedings of the 12th International Conference on CAD/CAM Robotics and Factories of the Future, Vol. , pp. , Middlesex University, London, 1996.

AUTOMATIC MACHINE SETTING FOR INJECTION MOLDING PROCESS

K.H. Tan and M.M.F. Yuen
Department of Mechanical Engineering
Hong Kong University of Science and Technology
Hong Kong

ABSTRACT

This paper describes the development of a simplified analytical model for the injection molding process. The model assumes an isothermal flow during filling and a static cooling during post-filling. The viscosity of the polymer melt is represented by a power law model and the thermal diffusivity is assumed to be a constant. The mold geometry is represented by the composition of several simple geometrical units. This model is applied to suggest a processing window and a set of initial processing parameters. A simulation case study is presented using a centre-gated disc mold with PS. Results show that the suggestion is in the right order of magnitude.

KEYWORDS

Injection Molding Process, Analytical Model, Quality Criteria, Processing Parameters

1. INTRODUCTION

A process model is very useful in setting the injection molding processing parameters. Modeling of the process has been studied extensively in the literature. Filling of simple geometries has been analyzed by many authors, for example Kamal and Kenig [1], and Williams and Lord [2]. The exact solution for cooling of plate and cylinder are given by Carslaw and Jaegar [3], and Biot [4] respectively. For the case of filling and post-filling (holding and cooling) of complicated geometries, definite works have been done by Hieber and Shen[5], and Chiang et al. [6][7], which resulted in a commercially available simulation package, C-MOLD. Attempts have been done by Tan and Yuen [8], Pandelidis and Zou [9], and Choi et al. [10] to determine the processing parameters by using such simulation packages. This approach involves creating a finite element model and running a number of simulations. These efforts may not be feasible in a shop-floor production environment. An alternative is to use an analytical model to determine the processing parameters instead of using these simulation packages. Although the processing parameters determined by using the analytical model is expected to be less accurate, it is compensated by a reduction of modeling effort. In this paper, the development of a simplified analytical model for the injection molding process is described. This model is applied to predict the processing window, and the processing parameters are determined based on this processing window.

2. QUALITY CRITERIA OF INJECTION MOLDING PROCESS

Injection molding process consists of three stages: filling, holding and cooling. In order to produce parts with acceptable quality, some necessary criteria have to be satisfied. During the filling stage, the machine capacity should be large enough to completely fill the mold with the polymer melt and keep the mold being closed. This can prevent short shot and flash. In addition, the shear stress and the shear rate should not be too large so as to prevent melt-fracture and degradation respectively. During the holding stage, the gate should be solidified before the release of holding pressure to prevent backflow of polymer melt from the mold cavity to the nozzle. During the cooling stage, the cooling time should be long enough for the molded part to gain the necessary rigidity for demolding. A set of necessary quality criteria can be defined based on the above discussion. They are:

- $P_{fill} < P_{inj}$
- $F_{cavity} < F_{clamp}$
- $\tau_{max} < \tau_{critical}$
- $\dot{\gamma}_{max} < \dot{\gamma}_{critical}$
- $T_{gate} < T_{solidify}$
- $T_{cavity} < T_{demold}$

where P_{fill} is the pressure required to fill the mold, F_{cavity} is the force generated by the cavity pressure, τ_{max} is the maximum shear stress, $\dot{\gamma}_{max}$ is the maximum shear rate, T_{gate} is gate temperature at the end of holding and T_{cavity} is the cavity temperature at the end of cooling. P_{inj} and F_{clamp} are the injection pressure and clamping force of the machine, $\tau_{critical}$ and $\dot{\gamma}_{critical}$ are the critical values for the onset of melt-fracture and degradation, and $T_{solidify}$ and T_{demold} are the solidifying temperature and demolding temperature.

3. ANALYTICAL MODELING OF INJECTION MOLDING PROCESS

The most important processing parameters for injection molding are the melt temperature (T_{melt}), filling time (t_{fill}), holding pressure (P_{hold}), holding time (t_{hold}) and cooling time (t_{cool}). For given mold and material, these parameters determine the values of the following quantities: filling pressure, cavity force, shear stress, shear rate and temperature at different time and locations. Injection molding process can be represented by a model with the processing parameters as the input, while the quantities related to the part quality as the output. The model can be developed analytically if some simplifying assumptions are made. The analytical modeling for mold filling and mold cooling is described in the following subsections.

3.1 Modeling of Injection Mold Filling

The following simplifying assumptions are made to model the filling of simple geometrical units,

1. The viscosity of the polymer melt obeys temperature-dependent power-law model.
2. The polymer melt flow is steady, one-dimensional, isothermal at the inlet melt temperature and at a constant flow rate.
3. The simple geometrical units are limited to long circular tube, wide and long rectangular channel, disc and sector of disc. (See Figure 1.)

Equation (1) shows the temperature-dependent power law viscosity model [11],

$$\eta = m(T)\dot{\gamma}^{n-1} = [B^n(\tau^*)^{n-1} exp(\frac{nT_b}{T})]\dot{\gamma}^{n-1} \qquad (1)$$

where η is the viscosity, m is the consistency index, n is the power-law index, B, τ^* and T_b are material dependent constants. The shear stress and shear rate for a power law fluid is related by Equation (2),

$$\tau = m(T)\dot{\gamma}^n \qquad (2)$$

The pressure drop equations for one-dimensional isothermal flows of simple geometrical units are available in [12]. These equations predict the pressure drop for a given flow rate. The predicted pressure drop can be used to calculate the maximum wall shear stress, and the maximum wall shear rate can be obtained by Equation (2). The equations for each simple geometrical unit are summarized in Table 1. More complicated mold geometries can be handled by composition of the simple geometrical units. Such an approach has been used before, for example by Agassant et al. [13]. In this study, this approach is used, but some restrictive assumptions are made. Firstly, it is assumed that a single-cavity molding may be approximately represented by a series composition of the simple geometrical units. Secondly, the multi-cavity moldings under consideration are naturally balanced. Thirdly, only single-gated cavity molding is considered. Since isothermal flow is assumed, the temperature at each simple geometrical unit is the same and equals to the inlet melt temperature. The consistency index, m, in each unit is also the same and is given by Equation (3),

$$m(T_{melt}) = [B^n(\tau^*)^{n-1} exp(\frac{nT_b}{T_{melt}})] \qquad (3)$$

For single-cavity molding, since the composition is assumed to be in series, the flow rates in the sprue, runner, gate and cavity are the same and equals to the overall flow rate as given in Equation (4),

$$Q(t_{fill}) = \frac{V}{t_{fill}} \qquad (4)$$

where V is the total volume of the molding. For multi-cavity molding, the flow rate in the sprue equals to the overall flow rate. If there are N_1 primary runners, then the flow in the sprue will be divided into N_1 branches and the flow rate in each primary runner will be given by Equation (5),

$$Q^1_{runner} = \frac{Q}{N_1} \tag{5}$$

The flow rates in the secondary runners (if any) are reduced similarly. The flow rates in the gate and in the cavity equal to that of the runner connecting to them. Based on the values of m and Q in each simple geometrical unit, the pressure drop, the maximum wall shear stress and the maximum wall shear rate can be calculated using equations summarized in Table 1. The filling pressure of the composite unit is the sum of all the individual pressure drops. The maximum shear stress and shear rate of the composite unit are the maximum values among all the simple units. These equations are shown as below,

$$P_{fill} = \Delta P_{sprue} + \Delta P_{runner} + \Delta P_{gate} + \Delta P_{cavity} \tag{6}$$

$$\dot{\gamma}_{max} = \max(\dot{\gamma}_{sprue}, \dot{\gamma}_{runner}, \dot{\gamma}_{gate}, \dot{\gamma}_{cavity}) \tag{7}$$

$$\tau_{max} = \max(\tau_{sprue}, \tau_{runner}, \tau_{gate}, \tau_{cavity}) \tag{8}$$

The composition scheme for the single-cavity molding and multi-cavity molding are illustrated in Figure 2. The cavity force at the end of filling can be estimated by integrating the pressure over the projected area of the molding. However, this estimation is too small as compared to the force generated during packing. A rough but more realistic estimation could be as follow,

$$F_{cavity} = N_{cavity} \times \Delta P_{cavity} \times A_{cavity} \tag{9}$$

where N_{cavity} is the number of cavity and A_{cavity} is the projected area of each cavity. This estimation assumes the cavity pressure is uniformly distributed at a value equals to the cavity inlet pressure and the force due to the delivery system is ignored.

3.2 Modeling of Injection Mold Cooling

The following simplifying assumptions are made to model the cooling of an injection molded part,

1. The molded part is at rest and there is no heat generation.
2. The thermal diffusivity (α) can be treated as a constant.
3. The cooling is unidimensional. That is, plates and cylinders are considered to be infinitely extending and infinitely long respectively.
4. The initial temperature of the part is uniformly distributed at T_{melt} and the boundary condition is isothermal at T_{mold}.

The exact solutions for the infinitely extending plate and the infinitely long cylinder are in the form of an infinite series [3][4]. Approximate solutions are possible if only the first term in the series is considered. The approximate solutions of the midplane temperatures ($\hat{T}$) of the plate and the cylinder are shown in Equations (10) and (11).

Plate:
$$\frac{\hat{T} - T_{mold}}{T_{melt} - T_{mold}} = \frac{4}{\pi}\exp\left(\frac{-\pi^2 \alpha t}{4h^2}\right) \tag{10}$$

Cylinder:
$$\frac{\hat{T} - T_{mold}}{T_{melt} - T_{mold}} = 1.5992\exp\left(-5.784\frac{\alpha t}{R^2}\right) \tag{11}$$

In some cases, average temperatures ($\overline{T}$) across the thickness are of interest, and they are shown in Equations (12) and (13).

Plate:
$$\frac{\overline{T} - T_{mold}}{T_{melt} - T_{mold}} = \frac{8}{\pi^2}\exp\left(-\frac{\pi^2 \alpha t}{4h^2}\right) \tag{12}$$

Cylinder:
$$\frac{\overline{T} - T_{mold}}{T_{melt} - T_{mold}} = 0.69156\exp\left(-5.784\frac{\alpha t}{R^2}\right) \tag{13}$$

4. PROCESS WINDOW AND INITIAL SETTING

The set of melt temperature and filling time satisfying the shear stress and shear rate criteria is defined as the processing window (W). That is,

$$W = \{ (T_{melt}, t_{fill}): \tau_{max} < \tau_{critical} \text{ and } \dot{\gamma}_{max} < \dot{\gamma}_{critical} \}.$$

The initial setting of melt temperature and filling time is selected from the processing window which minimizes the cycle time. The initial setting is the solution of the following constraint minimization problem:

$$\textbf{Minimize} \quad t_{cycle} = t_{fill} + t_{cool}$$
$$\textbf{Subjected to} \quad (T_{melt}, t_{fill}) \in W$$

where t_{cycle} is the cycle time, t_{cool} is the cooling time. The required filling pressure and clamping force are calculated at this initial set of melt temperature and filling time. The holding pressure is set to the 80% of the calculated filling pressure, as recommended in [14]. The holding time is defined as the time required for the midplane temperature of the gate passes through the solidifying temperature of the material [15]. For cooling time, the requirement that the midplane temperature should reach the demolding temperature is unnecessary restrictive [16], so average temperature is used instead of midplane temperature.

5. CASE STUDY

A simulation case study is presented using a centre-gated disc mold with PS (Diarex HT60) from Mitsubishi Chemical. The mold dimensions and the material data are shown in Table 2 and Table 3. Figure 3 shows the filling pressure, maximum shear stress and maximum shear rate predicted by the analytical model, with comparison to the prediction by C-MOLD. Note that the maximum shear rate predicted by C-MOLD shown here is the representative shear rate instead of the wall shear rate. The representative shear rate characterizes the magnitude of the shear rate at a cross section [14]. In general, the analytical filling pressure and maximum shear stress are smaller than that predicted by C-MOLD. The deviations are larger for longer filling time or lower melt temperature. It is because the cooling effect during filling is larger in these regions and the isothermal flow assumption may not be valid. For the case of maximum shear rate, the prediction by C-MOLD is smaller than the analytical prediction since prediction from C-MOLD is a representative value over a cross section, instead of a maximum value at the wall. Based on the comparison, it can be concluded that the prediction by the analytical model is in the right order of magnitude and the accuracy is higher in the regions of short filling time or high melt temperature. Based on the analytical model, a processing window and an initial setting of the processing parameters are suggested, as shown in Figure 4 and Table 4 respectively. A simulation is run in C-MOLD using these parameters and the results are shown in Table 5, with comparison to the values predicted by the analytical model. The analytical filling pressure and maximum wall shear stress are smaller than the predicted values by C-MOLD. It is because the cooling effect in the suggested filling time and melt temperature is not negligible. The suggested clamping force is also smaller than that predicted by C-MOLD since the pressure in the latter case is larger. The analytical average temperature in the cavity and that predicted by C-MOLD are shown in Figure 5. The C-MOLD prediction shown here is the average temperature of an element in the cavity. It should be noted that, theoretically, the initial analytical average temperature should be at 200°C (See assumption 4 for mold cooling). The deviation from the theoretical case is due to the first term approximation in the exact solution, but the error is significant only at a very short cooling time. It can be observed that the average temperature predicted by C-MOLD is lower than the analytical prediction, especially in the initial phase of cooling. This deviation is due to the cooling effect during filling, and this effect is less significant as cooling continues. There is another point worth mentioning. As shown in Table 4, the holding time is the same as the cooling time. It is because the dimension of the sprue gate is larger than that of the cavity, and the cooling time for the sprue gate is longer. As the mold is cooled for 8 seconds, cooling in the cavity is already sufficient and the holding pressure can be released although the sprue gate is not fully solidified.

6. CONCLUSIONS

A simplified analytical model is developed for the injection molding process. This analytical model is applied to predict the processing window and the initial setting of the processing parameters. A case study is presented using the disc mold with PS. Results show that the prediction is in the right order of magnitude as comparing with the prediction by C-MOLD.

7. FIGURES

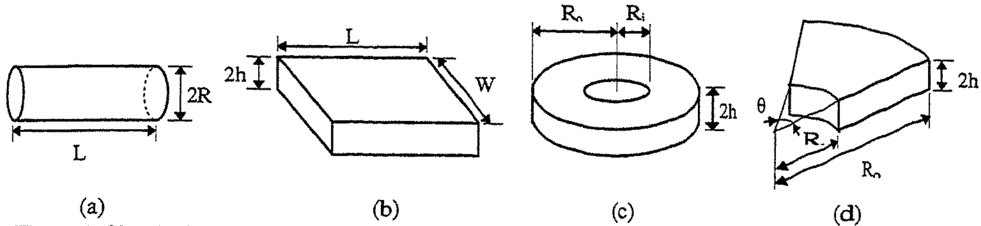


Figure 1. Simple Geometrical Units. (a) Circular Tube. (b) Rectangular Channel. (c) Disc. (d) Sector.

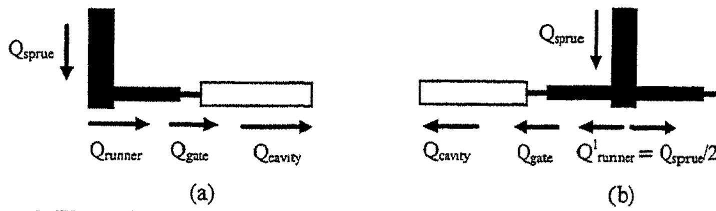


Figure 2. Illustration of Composition Scheme. (a) Single-cavity Molding. (b) Multi-cavity Molding.

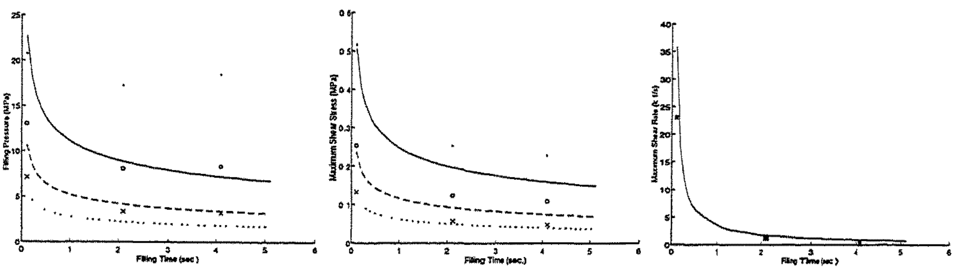


Figure 3. Analytical Model and C-MOLD Results Comparison. (a) Filling Pressure. (b) Maximum Shear Stress. (c) Maximum Shear Rate. Line: Analytical Model. Point: C-MOLD. T_{melt} = 200°C: "—" and "•". T_{melt} = 250°C, "---" and "o". T_{melt} = 300°C, "...." and "x".

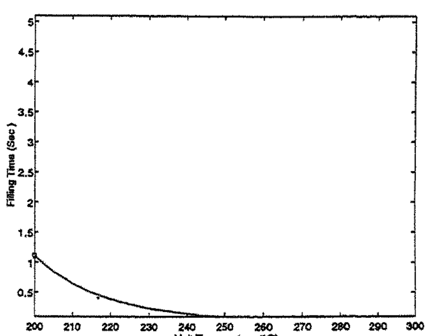


Figure 4. Processing Window (above solid line) and Initial Processing Parameters (shown by "o").

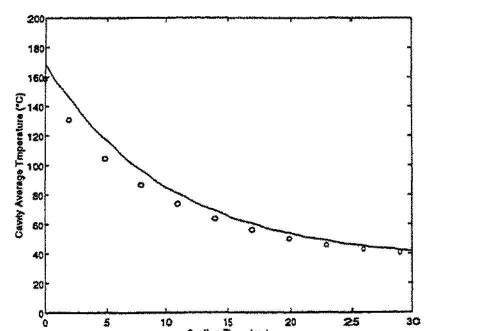


Figure 5. Average Temperature in Cavity. Solid line: Analytical Model. "o": C-MOLD.

8. TABLES

Geometry	ΔP	τ_{max}	$\dot{\gamma}_{max}$
Circular Tube	$\frac{2mL}{R}[\frac{(3n+1)Q}{n\pi R^3}]^n$	$m[\frac{(3n+1)Q}{n\pi R^3}]^n$	$\frac{(3n+1)Q}{n\pi R^3}$
Rectangular Channel	$\frac{mL}{h}[\frac{(2n+1)Q}{2nWh^2}]^n$	$m[\frac{(2n+1)Q}{2nWh^2}]^n$	$\frac{(2n+1)Q}{2nWh^2}$
Disc	$\frac{m}{(1-n)h}[\frac{(2n+1)Q}{4n\pi h^2}]^n(R_o^{1-n}-R_i^{1-n})$	$m[\frac{(2n+1)Q}{4n\pi R_i h^2}]^n$	$\frac{(2n+1)Q}{4n\pi R_i h^2}$
Sector	$\frac{m}{(1-n)h}[\frac{(2n+1)Q}{2n\theta h^2}]^n(R_o^{1-n}-R_i^{1-n})$	$m[\frac{(2n+1)Q}{2n\theta R_i h^2}]^n$	$\frac{(2n+1)Q}{2n\theta R_i h^2}$

Table 1. Pressure drop, maximum shear stress and maximum shear rate for simple geometrical units.

Sprue Gate		Disc Cavity		
L = 50 mm	R = 3.25 mm	R_i = 3.25 mm	R_o = 80 mm	h = 1.5 mm

Table 2. Centre-gated Disc Mold Dimensions.

Viscosity Model	n = 0.30981	τ^* = 29479 Pa	B = 6.05×10^{-8} Pa s	T_b = 12113 K
Thermal Data	$\alpha = 0.89\times10^{-7}$ m^2/s			
Quality Data	$\tau_{critical}$ = 0.24 MPa	$\dot{\gamma}_{critical}$ = 40000 1/s	$T_{solidify}$ = 96°C	T_{demold} = 96°C

Table 3. Material Data of PS (Diarex HT60) from Mitsubishi Chemical. α is from [17]. $\tau_{critical}$ and $\dot{\gamma}_{critical}$ are from [14].

T_{melt}	t_{fill}	P_{hold}	t_{hold}	t_{cool}
200 °C	1.1 sec.	11 MPa	8 sec.	8 sec.

Table 4. Initial Processing Parameters.

	P_{inj} (MPa)	F_{clamp} (Ton)	τ_{max} (MPa)	$\dot{\gamma}_{max}$ (1/s)	$\bar{T}_{cavity}$ (°C)
C-MOLD	17	17	0.25	2118	86
Analytical Model	13	12	0.24	3250	96

Table 5. Analytical Model and C-MOLD Results Comparison at Initial Processing Parameters.

9. REFERENCES

[1] Kamal, M.R. and Keing, S., "The Injection Molding of Thermoplastics. Part I: Theoretical Model", Polymer Engineering and Science, Vol.12, No.4, pp.294-301, 1972.

[2] Williams, G. and Lord, H.A., "Mold-Filling Studies for the Injection Molding of Thermoplastic Materials. Part I: The flow of Plastic Materials in Hot- and Cold-Walled Circular Channels", Polymer Engineering and Science, Vol.15, No.8, pp.553-568, 1975.

[3] Carslaw, H.S. and Jaeger, J.C., Conduction of Heat in Solids, Oxford University Press, 1959.

[4] Biot, M.A., Variational Principles in Heat Transfer, Oxford, London, New York, 1970.

[5] Hieber, C.A. and Shen, S.F., "A Finite-Element/Finite-Difference Simulation of the Injection Molding Filling Process", Journal of Non-Newtonian Fluid Mechanics, Vol.7, No.1, pp.1-32, 1980.

[6] Chiang, H.H., Hieber, C.A. and Wang, K.K., "A Unified Simulation of the Filling and Postfilling Stages in Injection Molding. Part I: Formulation", Polymer Engineering and Science, Vol.31, No.2, pp.116-124, 1991.

[7] Chiang, H.H., Hieber, C.A. and Wang, K.K., "A Unified Simulation of the Filling and Postfilling Stages in Injection Molding. Part II: Experimental Verification", Polymer Engineering and Science, Vol.31, No.2, pp.125-139, 1991.

[8] Tan, K.H. and Yuen, M.M.F, "Processing Parameter Setting for Injection Molding of Plastic Parts", Simulation of Materials Processing: Theory, Methods and Applications - NUMIFORM 95, Eds. Shen, S.H. and Dawson, P., pp.1197-1202, A.A. Balkema, 1995.

[9] Pandelidis, I. and Zou, Q., "Optimization of Injection Molding Design. Part II: Molding Conditions Optimization", Polymer Engineering and Science, Vol.30, No.15, pp.883-892, 1990.

[10] Choi, G.H., Lee, K.D., Chang, N. and Kim, S.G., "Optimization of Process Parameters of Injection Molding with Neural Network Application in a Process Simulation Environment", Annals of CIRP, Vol.43, No.1, pp.449-452, 1994.

[11] Hieber, C.A., "Melt-Viscosity Characterization and Its Application to Injection Molding", Injection and Compression Molding Fundamentals, Eds. Isayev, A.I., pp.1-136, Marcel Dekker, Inc., 1987.

[12] Lee, C.C. and Castro, J.M., "Model Simplification", Fundamentals of Computer Modeling for Polymer Processing, Eds. Tucker III, C.L., pp.69-112., Hanser Publishers, Munich Vienna New York, 1989.

[13] Agassant, J.F., Alles, H., Philipon, S. and Vincent, M., "Experimental and Theoretical Study of the Injection Molding of Thermoplastic Materials", Polymer Engineering and Science, Vol.28, No.7, pp.460-468, 1988.

[14] AC Technology, C-MOLD Reference Manual & User's Guide, 1995.

[15] Greener, J., "General Consequences of the Packing Phase in Injection Molding", Polymer Engineering and Science, Vol.26, No.12, pp.886-892, 1986.

[16] Liang, J.Z. and Ness, J.N., "The Calculation of Cooling Time in Injection Moulding", Proceedings of the International Conference on Manufacturing Automation, Eds. Ko, N.W.M., and Tan, S.T., pp.830-835, 1992.

[17] Menges, G. and Mohren, P., How to Make Injection Molds, Hanser Publishers, Munich Vienna New York, 1986.

DEVELOPMENT OF AN INTELLIGENT ENTERPRISE INFORMATION SYSTEM

Johnny, K.C. Ng and W.H. Ip
Department of Manufacturing Engineering
The Hong Kong Polytechnic University
Hong Kong

ABSTRACT

One of the crucial requirements of global manufacturing strategy is the ability to optimise the use of information systems. In order to achieve this, a manufacturing information system must reflect how work is done, to support the overall business objectives, without limited by the hardware configuration and software capability. Because of different business functions, every organisation has an unique manufacturing information system requirement which cannot be addressed by traditional systems. This paper proposes an architecture for an Intelligent Enterprise Information System (IEIS) for a global manufacturing environment. The implementation was designed with a Distributed Object-Oriented Technology (DOOT) which increases the system scalability, configurability, flexibility and interoperability. The Intelligent Enterprise Information System (IEIS) is designed to support real-time data collection and processing for all required production and inventory analysis. The IEIS is designed with a flexible architecture which can meet the distributed business requirements.

KEYWORDS

Information System, Object-Oriented Technology, Client/Server Architecture.

1. INTRODUCTION

With the increasing demand of high technology in manufacturing industry, a higher degree of market competition and globalisation, and the transformation of the work environment towards information based offices in the developing countries and newly industrialised economics, the importance of a flexible and effective information system is becoming significant. In the last 10 to 20 years the reducing cost and increasing capability of computer software and hardware has apparently given the manufacturing industry a tool with which to meet the increasing demand [1]. However, information technology (IT) has often been installed on a piecemeal tactical basis, meeting the needs of a particular function rather than the enterprise as a whole.

Globalisation has affected manufacturing sectors in Hong Kong since many of their production facilities are located in China and other countries and their logistics function spread across various countries in different physical locations. Enterprises participating in these global markets will increasingly be at a serious strategic disadvantage if they are unable to handle their world-wide operations and manage them in a globally co-ordinated manner. Therefore, they must look at the issues and opportunities of using a scaleable information system on a distributive basis.

ERP (Enterprise Resources Planning) represents the application of newer information technology to the MRP II model. These technology changes include the move to relational database management systems (RDBMS), the use of a graphical user interface (GUI), open systems and a client/server architecture [2]. The new client/server technology makes the processing power of PC (personal computer) comparable to midrange systems as well as mainframes. Open system architecture allows companies to build distributed applications that provide true ERP and COMMS (Customer Oriented Manufacturing System) capabilities [3]. The low cost of communication technology, combined with configurable Local Area Networks (LANs) and Wide Area Networks

(WANs), enable flexible, scaleable, distributed system, as well as Computer Integrated Manufacturing (CIM) to become a reality.

Manheim [4] and Saxena & Ma [5] argue that a global-competing enterprise is influenced by the following global forces:

- round-the-clock trading in tightly-linked global financial markets;
- global sourcing of inputs, marketing and distributed of products, and manufacturing of components and final products;
- increased pressures for improved product quality and reduced product price; evolution of business toward more comprehensive and continuos global co-ordination.

In order to sustain the competitive advantage of local manufacturing industry, the effective integration and globalisation of manufacturing are major issues to be addressed. A sound enterprise-wide information system is considered to be an essential tool for adapting this global environment. In this paper, the authors propose an architecture of Intelligent Enterprise Information System (IEIS) with high scalability and configurability in order to achieve the global manufacturing strategy.

2. REQUIREMENTS ON AN ENTERPRISE INFORMATION SYSTEM

The continuing expansion of transnational economic activities and the market globalisation [6] have in recent years resulted in a broadened scope of competition strategies [7]. In order to overcome the size limitation of the domestic market, many organisations are expanding into markets far beyond the borders of the territory. The land and labour cost of Hong Kong are very high today, leading many manufacturing companies have moved their production facilitates to other locations. This global activity necessitates establishing operations in different countries that may use different character sets, speak different languages, and have different laws, customs, and currencies. In this information-oriented society, the need of an enterprise information system for global manufacturing environment becomes dominant.

The overall goal of information technology within a manufacturing environment is to help achieve the business goals of an enterprise. Information technology may be viewed as an adaptive mechanism for following the frequent changes and evolution of the technological environment, the market, and organisational structures. Four key aspects of an information system are; (i) Scalability, (ii) Configurability, (iii) Flexibility, and (iv) Interoperability.

Furthermore, the system architecture must support enterprise that vary dramatically in size. A fundamental requirement of a sound information system is that it be scalability and configurability so it can be easily adapted the dynamic environment.

This Global information system requires an architecture which cannot be met by traditional system design approach [8]. In traditional systems design we define the informational (data) and behavioural (functional) element separately, and in doing so clearly distinguish between information and functions. Information is typically stored in a shared or distributed database which can be accessed by applications. Applications are implemented in individual programs. The traditional approach of building applications creates a number of functional dependencies, thus reducing flexibility for the introduction of changes [9]. A novel approach to system design must be investigated.

3. A DISTRIBUTED OBJECT ARCHITECTURE FOR INTELLIGENT ENTERPRISE INFORMATION SYSTEM

Objects can represent organisations, incidents, or roles which individuals or organisations play [10]. Object-Oriented Technologies (OOT) also employ the principle of inheriting characteristics or attributes from super class objects. The inheritance mechanism of OOT supports reusability of software, and simplifies design. Many literature [11,12,13] mentioned the employment of object-oriented approach to model and develop manufacturing systems.

A novel architecture for Intelligent Enterprise Information System (IEIS) is proposed. This architecture adopts the Distributed Object-Oriented Technology (DOOT) which aims to enhance the scalability and configurability of the system. This architecture is divided into four different domains; (i) Presentation Domain (PD), (ii) Reporting Domain (RD), (iii) Logic Domain (LD), and (iv) Database Domain (DD).

One of the characteristics of this architecture is that presentation, reporting, logic and database can be placed in different platforms and/or machines in a configurable manner as presented in figure 1. It enables the reduction of network traffic and enhances the system flexibility.

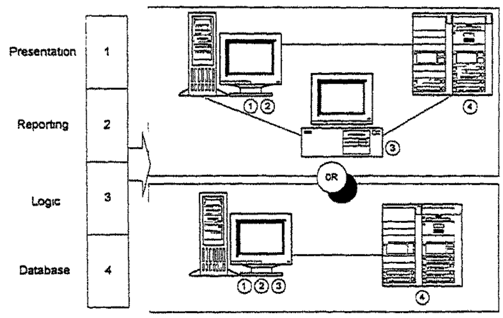


Fig. 1: Configurable System Architecture for Intelligent Enterprise Information System

The design of IEIS animates the people, places, and things that exist in the Enterprise in order to simulate the enterprise operation. The processing logic encapsulating these entities controls the operation of the enterprise and records events as they occur. The people, places, and things are presented to the desktop as icons and document windows. A Business Object is the visual image and the indigenous behaviour of the thing it represents. Nevertheless, a Business Object is the conceptual image of the job-related object that the user deals with everyday.

The model is divided into two distinct layers. They are Real Object layer that defines the user's conceptual view and Application layer that defines the corporate resources view. The Real Object layer belongs to the Presentation Domain while the Application layer exists in the Logic Domain.

The Presentation Domain is concentrated on the interaction with the user and managing the user's interface. The Reporting Domain is responsible for preparing of any report, graph, or query. The Logic Domain is focused on managing the corporate resources (e.g. database tables) that users share, in particular, the application logic. Only the Logic Domain can interact with the Database Domain. The Database Domain is a Relational Database (RDB) server (e.g. Oracle, Informix, SQL Server, etc.) that manages the persistent storage of data, however, this domain will not be discussed thoroughly in this paper. The following sections elaborate the key elements of the proposed system architecture.

3.1 Business Object Models

Business Object Model is defined by Object Management Group [14] as a representation of real business world entity. For example, it can be a customer, purchasing order, or a finished product. The user's conceptual model of a Business Object is not a direct reflection of the Application model. The Application model is represented as an Entity/Relationship (E/R) Model. The E/R Model is a master blueprint defining all of the business entities, relationships, and attributes that must be maintained to preserve the integrity of the corporate information resources. Figure 2 represents an architecture of the system.

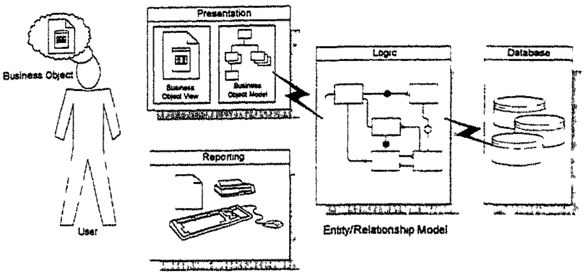


Fig. 2: A Distributed Architecture for the System

The Business Object Model is closer to the user's concept of the data representation than the E/R Model, and showing only the portion of the master blueprint that is relevant to the business activity in which the user is engaged. The conceptual model of a Business Object takes the shape of a hierarchical composite object that is identified by the root subject entity. For example, the Purchase Order is a form to fill out which has some header information and line items but the E/R Model will define a Purchase Order as the combination of multiple entity types with connections to many related entities.

3.2 Presentation Domain (PD)

Presentation Domain consists of three application components. They are Models, Views, Elements. The architecture is presented as figure 3.

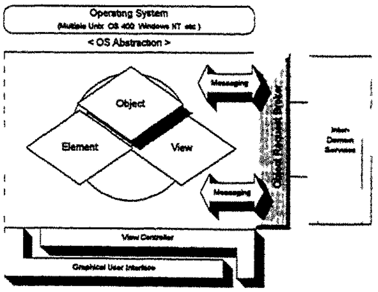


Fig. 3: Presentation Domain

The Model is the central control point of a business object. It represents a hierarchical data collection, with a root object that identifies the business object. Model contains subordinate subjects which is a list of component entities. It manages the data cache, requests data from servers, updates the servers by sending transactions, and interacts with the user through the View. Each business Entity/Relationship represented in the Model is implemented as a server object in the Logic Domain (LD). The Business Object Model reflects the concept of an object in the workplace that user can directly manipulate, including changing or adding views of an object and dragging and dropping a graphic business object on other business object to exchange data.

A Business Object View is a Graphical User Interface (GUI) window displaying information contained in the model. View is the controller of the window and widgets in the display. It responds to actions taken by the user, for instant, stroking the keyboard or clicking the mouse. View uses simple logic that can be executed without interacting with the server. The interaction of the logic and server will only perform by the Model, and the action will only be taken by user requisition. An ASCII file, which is described in figure 4, called View Definition File (VDF) predefines the View layout. Element is responsible for checking the validity and correct format of entered value in each widget. If the user enters an invalid character, the system will not echo the keystroke in the field and sound an error.

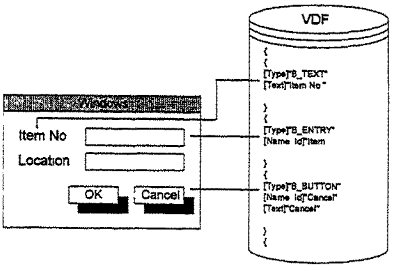


Fig. 4· A simplified representation of View Definition File

3.3 Logic Domain (LD)

The functions of Logic Domain are to manage the corporate database, maintain its integrity in order to maximise the usefulness of it, and provide program logic to cater for the business operation. This domain is designed to retrieve data in a variety of useful ways and update the database in a safe and controlled manner. It also ensures that the event-driven process are triggered and the data users possess on their workstations is up-to-date. The application components of Logic Domain are Entities, Relationships, Elements, Transactions, and Databases. The architecture is presented as fig. 5.

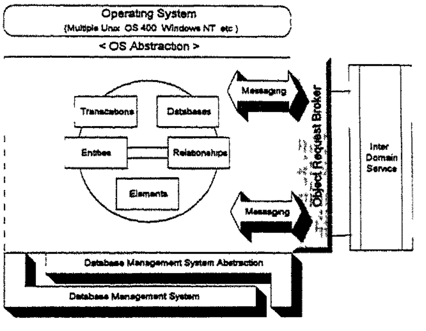


Fig. 5: Logic Domain

The implementation of E/R Model is the backbone of the Logic Domain architecture. The Entities object and Relationships object receive a message, process it, and return a reply. The hierarchical object model is shown as figure 6.

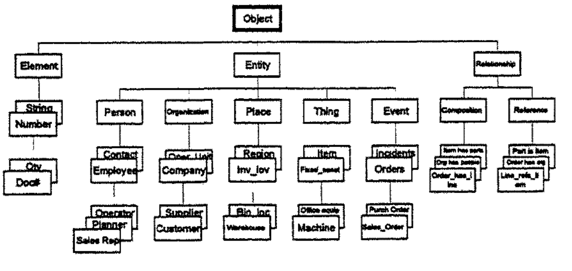
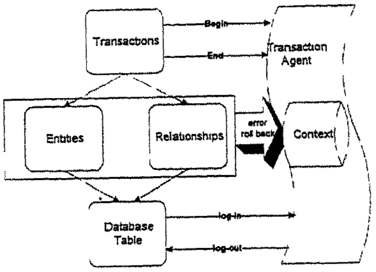


Fig. 6 : Hierarchical Object Model

Relationship in Logic Domain is mainly responsible for spreading the event driven business process. It also provides a query method that will produce a list of member of the Relationship. The query specifies which column to include, ordering, criteria and volume specification. Moreover, the Relationship can assign the sequential key to Entity by query the database. The Relationship is also used to resolve concurrency locks and register interest.

Transaction is the controller of all database updating activities. The function of a Transaction object is to ensure that the database is always recoverable to a logically consistent state. The transaction agent can be aborted at anytime and the updates can be rolled back when error signal detected by any object. The operation of Transaction is represented as figure 7.




```
Send Log-in message to the Transaction Agent.
 Send Lock message to the root of the unit-of-work
  For each update operation
  BEGIN
    Send update operation to its respective Business Entity Manager
  END
 Send Unlock message to the root of the unit-of-work
Send Log-out message to the Transaction Agent
```

Fig. 7: The logical flow of Transactions

3.4 Reporting Domain (RD)

Reporting Domain is responsible for preparing of any report, graph, or query. This domain is comprised of three components, namely, presenter, extractor, and analyser. Presenter is user to present all the report, graph, or query to the user by all available media (e.g. screen, printer). Extractor can communicate with Logic Domain through a Database object to call the Database Management System (DBMS) to perform all SQL operations. Analyser assimilates the data and prepares it for presentation. All of the calculation and sorting operations for a report are performed by this component.

4. THE IMPLEMENTATION

The application of distributed object-oriented technology in the development of an Intelligent Enterprise Information System is considered to be an effective approach for handling the system complexity. An information system for global manufacturing environment must be a very large system and it must be very demanding on the scalability, flexibility, configurability and inter-operability. Moreover, many of the modules are reusable across the system and many similarities exist in many components. Therefore, the object-oriented approach is the comprehensive methodology to design and develop a system.

The IEIS can be developed by using "traditional" object-oriented design methodology such as Booch [15]. However, distributed objects as defined by Islam [16] can be easily used to maintain the structure and behaviour of the objects in distributed manner.

Since the Business Object Model is the simulation of real world entity, it must take a huge effort to identify and implement. The long development life cycle is identified as main consideration of this object-oriented approach. It is being resolved by using the advanced Object-Orient Programming (OOP) language.

5. CONCLUSION

The Global pressure for manufacturing industry accelerates the development of advanced information system with flexible architecture. A distributed Object-Oriented architecture for Intelligent Enterprise Information System (IEIS) is proposed in this paper. The design adopting Distributed Object-Oriented Technology (DOOT) aims for the replacement of current rigid architecture in the manufacturing industry to provide more flexibility, scalability, configurability and interoperability. The architecture is divided into four different domains; (i) Presentation, (ii) Reporting, (iii) Logic, and (iv) Database, that can be configured in the network with distributed manner to meet the global manufacturing requirements. Further efforts are being carried out in the identification and the development of Business Object Model and the implementation of a prototype system.

6. REFERENCES

1. Sotiris K. and Harold C., "Developing Strategic Information Systems", Integrated Manufacturing System, Vol. 5, No. 2, pp.4-11, 1996.
2. Robert Gumaer, "Beyond ERP and MRP II", IIE Solution, pp.32-35, September 1996.
3. Ray Martin, "Turbocharging MRP II Systems with Enterprise Synchronisation", IIE Solution, pp.32-34, November 1995.
4. Manheim, M. L., "Global information technology : Issues and opportunities", Proceeding of the 24th Hawaii International Conference on System Sciences, 4, 172-181, 1991.
5. Saxena K.B.C. & Ma L.C.K., "Globalisation in IT Education", Global Information Technology Education : Issues and Trends, Ideal Group Publishing, USA, pp.282-309, 1993.
6. Shiller, H.I., "The Erosion of National Sovereignty by the World Business System", The Myth of the Information Revolution-Social Ethical Implications of Communication Technology, SAGE Communications in Society Series, pp.21-34, 1986.
7. Kotler, P., Marketing Management-Analysis, Planning, Implementation and Control (6th ed.), Prentice-Hall International Editions, Englewood Cliffs, NJ, 1988.
8. Bals, P., "An Object Oriented Analysis and Design Methodology for Computer Integrated Manufacturing Systems", Proceedings Tools Conference, pp.75-84, 1989.
9. Bauer, Bowden, Browne, Duggan and Lyons, Shop Floor Control Systems, Chapman & Hall, 1994.
10. Bernard P. Zeigler, Object-Oriented Simulation with Hierarchical, Modular Models, Academic Press, 1990.
11. A. Sherif EL-Gizawy, and Mitar Thunga, "Object-Oriented Model for Automatic Generation of Assembly Plans" , IFAC Intelligent Manufacturing System, P. Kopacek, pp.525-531, 1994.
12. H. Stienen, P. R. van der Weerd, "Object Oriented Design of a Flexible Manufacturing System", Knowledge based production management systems, J. Browne, pp;329-339, 1989.
13. Ji, Ping, Manufacturing data representation using an object-oriented data model, Ph.D. Dissertation, 1991
14. OMG, Object Analysis and Design, March 1994.
15. Booch, G., Object-Oriented Design, Benjamin/Cummings, 2nd Edition, 1994.
16. Lslam, N., Distributed Objects : Methodologies for customising system software, IEEE Computer Society Press, 1996

A SIMULATION BASED OSCILLATORY SEQUENCING ALGORITHM FOR LAYOUT PLANNING OF FMS

V.H.R. Subrahmanyam and N. Ramesh Babu
Manufacturing Engineering Section
Department of Mechanical Engineering
Indian Institute of Technology
Madras - 600 036
India.

ABSTRACT

This paper addresses the Layout Planning problem, one of the important design issues of Flexible Manufacturing Systems (FMS). A simulation based oscillatory sequencing algorithm is developed to obtain the best possible layouts with an objective of minimising the material handling cost. The algorithm consists of four routines: Input, Sequencing, Layout Constructor, Evaluator. The Input routine accepts the number of work-stations, their interaction matrix and the sizes of the machines in each work-station as input from the user. The Sequencing routine generates the sequence of work-station identification numbers using biased random sequencing taking the interaction between the work-stations into consideration. The Layout Constructor generates a layout such that continuity in the department is ensured. Here, an X-Y double oscillatory sequencing algorithm is used. The Evaluator calculates the cost of each layout using the center of gravity of each department for the distance calculation and gives out the best possible layouts. Finally the algorithm gives out the fifteen best possible layouts as the output keeping the objective as minimisation of material handling cost.

KEY WORDS

Flexible Manufacturing Systems (FMS), FMS-Design, Layout Planning, Simulation.

1. INTRODUCTION

Greater product proliferation, market fragmentation and shorter product life cycles have made firms increasingly aware of the importance of manufacturing flexibility. Flexible Manufacturing Systems (FMSs) are becoming increasingly popular on account of their flexibility, adaptability and higher productivity. These systems are highly complex, expensive and pose many implementation problems. Stecke [1] classifies FMS implementation problems into four categories: 1) Design, 2) Planning, 3) Scheduling, and 4) Control. Among them, the issues such as planning, Scheduling and control have been addressed by several researchers whereas the FMS design issues have not received much attention. But, in many cases FMS managers struggle with scheduling problems under unrealistically tight capacity constraints due to inefficient designs.

The FMS design problems can be further classified into 1) Optimal System Configuration which involves the determination of number of machines to be installed in the system, 2) Specification of FMS Layout, 3) Selection of a Storage System that involves the determination of buffer sizes, 4) Specification of the type and capacity of Material Handling System (MHS) and 5) Determination of other important system resources like number of pallets, number and types of fixtures, number and types of tools etc. Among these problems, design of the physical layout of an FMS is of tremendous importance for the effective utilisation of the system [2]. Tompkins et al.[3] emphasised the importance of layout decisions for effective material handling by pointing out that 20 to 50% of the total operating expenses in manufacturing are attributed to material handling and layout related costs. Hence, the Layout Planning problem is considered in the present work.

2. BACKGROUND

In general the plant layout problems are formulated as Quadratic Assignment Problem (QAP) [4]. The QAP approach is efficient only for small sized problems because of the computational complexity involved. After Sahani et al [5] showed that the QAP is NP-Complete, the researchers have concentrated on heuristic algorithm development [6,7]. Heragu et al [8] have questioned the applicability of QAP formulation for FMS because FMS configuration violates the assumption of QAP. Bazaraa [9] discussed Quadratic Set-covering Problem (QSP) for FMS layout design and proposed a branch and bound approach. Kouvelis [2] emphasised that the QAP and QSP formulations ignore interactions between the layout decisions and the queuing performance measures of an FMS. Later on, some researchers have solved the problem using some Mathematical and analytical models [4,10,11,12].

Most of the existing algorithms except CRAFT and CORELAP are not designed to solve layout problems in which the facilities are of unequal area. However, the CRAFT and CORELAP can not be used for the FMS machine layout problem because the shapes of the machines may be altered in the final layout [13]. Elmaraghy [14], Rathmill et al [15], Carrie et al [16] and Martin et al [17] have emphasised the use of simulation for screening of alternate layouts. Zoller et al [18] emphasised that the layout constructor should generate a high percentage of feasible layouts with a minimum of computer time. There are three types of movements that are well suited for computer coding and high speed execution to obtain a layout: parallel, single- and double - oscillatory. Out of them, only the two modes of double oscillatory matrix traversal were reported to be performing satisfactorily in terms of all the above requirements. Hence, in the present work, a simulation based X-Y oscillatory sequencing algorithm is developed for Layout Planning of FMS.

3. METHODOLOGY

In the present work, a simulation based Oscillatory Sequencing Algorithm is developed for obtaining fifteen best possible layouts. The objective considered here is to reduce the material handling cost because the material handling cost is considered to be the main contributor to the overall manufacturing cost. The algorithm is divided into 4 routines: Input Routine, Sequencing Routine, Layout constructor and Evaluator. The flow chart in Fig. 1 illustrates the basic structure of the simulation model which was designed to implement this approach.

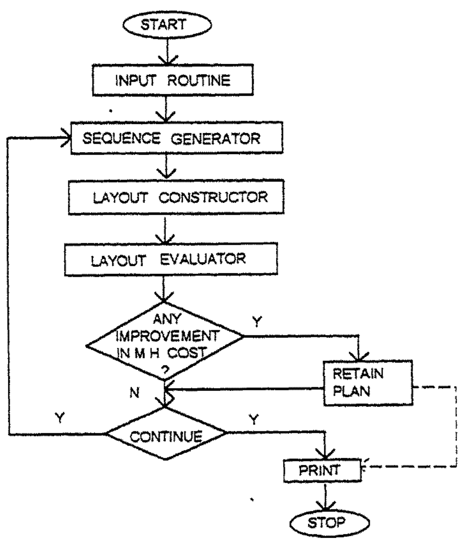


Fig. 1: Flow chart representing the Simulation model for Layout Planning.

3.1 Input Routine

This routine accepts the input on the configuration of the system along with the corresponding space requirements. It includes the number of machines in each work-station along with their sizes, clearance required for Material Handling System, and the interaction matrix between different work-stations. This information can be directly obtained from the object oriented simulation algorithm proposed for the design of FMS[19]. There exists a facility to fix a particular work-station at a particular place depending on the requirements. This is done by reserving the particular locations on the sample layout grid. If any particular area of the grid are required to be left vacant, dummy location fixing can be done. The dummies constrain the layout construction without entering the simulation or evaluation process.

3.2 Sequencing Algorithm

This routine generates the allocation sequence of the work-station identification numbers which decides the order in which they are fitted into the ground plan. Allocation sequences are provided by the simulator, a computerised pseudo-random generator. Since the mean biased random equivalent is much better than the best random observation [18], biased random sequencing is employed. The heuristic behind the biasing is taken as "The higher the flow volume between any two work centers, the more desirable their close proximity". As the simulator gives a different sequence for each run, a new layout is formed each time. The sequencing algorithm employed in the present work is given below.

Step 1: Set counter to 1.
Step 2: Get p, a random number between 1 and n, where n is the number of work-stations. Include p as the first element in the sequencing array.
Step 3: Get the interactions of all the non-included departments with department p.
Step 4: Take q, the largest of these values. Include the corresponding work-station identification number as the next element in the sequencing array.
Step 5: Increment counter by 1.
Step 6: If counter < n, Go to step 3;
else End.

3.3 Layout Constructor

This develops a layout based on the allocation sequence provided by the sequencing algorithm. A layout plan is called feasible only when it meets the specified requirements, assigns a continuous space of x_i moduli to each work center i. Planning economy dictates that the layout constructor mode should generate a high percentage of feasible layouts with a minimum of computer time. These conditions demand a highly systematic construction module, a class of which the three types of movements: parallel, single and double-oscillator that are well suited for computer coding and high speed execution. Among them, only the two modes of double oscillatory matrix traversion were found to perform satisfactorily in terms of all the above requirements [18]. The functioning of xy-oscillation is illustrated in Fig. 2, where only work-centers 1, 2, 5 and 6 are shown.

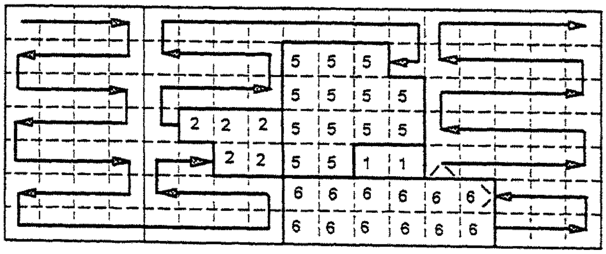


Fig. 2: (X,Y)- Oscillatory Construction Method.

Oscillatory construction modes require a sub division of the allocation matrix into sections consisting of several rows or columns each as shown in Fig. 2. The width of each section determines the maximum width or depth of all work-stations placed in it. Zoller et al.[18] have conducted experiments with several widths and suggested the section width to be taken as the rounded square root of the mean width. However, in the present algorithm, a provision is also made for the user to enter the section size.

Two problems may arise from the use of sections. First, an overflow may occur when the last work-station allocated to one section must be carried over into the adjacent section because of insufficient space in the former. Work center 6 in Fig. 2 shows how this problem is solved. The complete department is extended in its full depth into the adjacent section before the oscillatory routine takes over again. The second problem is that dummies and fixed departments may become obstacles in the path of the allocation process limiting the continuity of the allocation. If disrupted, the process will split departments into disjoint fragments and renders the whole layout infeasible. These disruptions can occur as a result of several set of conditions. The three generally encountered conditions are shown in Fig. 3 together with appropriate remedial action.

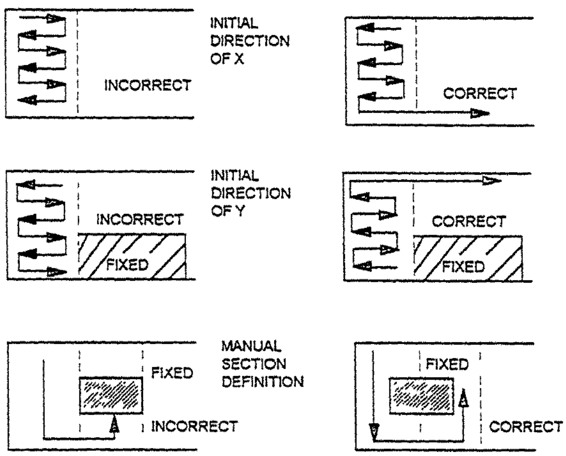


Fig. 3: Three Measures to assure continuous allocation.

3.4 Evaluator

This routine is used to evaluate the layout and to determine the best possible layouts with the objective of minimising the material handling cost.. Since the work-centers are two dimensional, it is necessary to represent them by points in order to measure the distances between work-stations. For this purpose center of gravity is taken as the reference point for this location. Center of gravity is computed from the co-ordinates of the cells belonging to each work center.

The evaluation routine computes the Total Material handling Cost, (TMC) as a linear function of Flow volumes per unit time between i^{th} and j^{th} work-stations. V_{ij}; and measured distance d_{ij} between i^{th} and j^{th} work-stations using Eq. (1).

$$TMC = \sum_{i=1}^{n} \sum_{j=1}^{n} V_{ij} d_{ij} \qquad (1)$$

4. RESULTS & DISCUSSIONS

A system with five work-stations having 3,4,3,2,4 number of machines in first, second, third, fourth and fifth work-stations respectively is considered for illustrating the present algorithm. The size of each machine is assumed to be the same. The interaction between the five work-stations is shown in Table - I.

	Interaction Priorities				
W/S No.	1	2	3	4	5
1	0	8	9	6	7
2	8	0	8	9	5
3	9	8	0	7	8
4	6	9	7	0	9
5	7	5	8	9	0

Table - I: The Interaction between different work-stations.
(0 - No Interaction, 9 - High Interaction)

The minimum cost layout is shown in Fig. 4 out of the best possible layouts obtained from the current algorithm, where the first letter in each cell indicates the work-station number and the second indicates the mchine number. As Zoller. et al.[18] points out, there is no need for Layout plans to be immediately applicable as blue prints for construction work, but rather should be sufficiently realistic to make minor changes. the generated layout plan can be used as a base to finalise the layout. Here, fifteen best possible layouts are generated to enable one to make minor changes as per the qualitative requirements of the user.

41	41	41	42	42	42				
41	41	41	42	42	42	33	33	33	33
54	54	54	21	21	21	33	33	32	32
54	54	54	21	21	21	32	32	32	32
53	53	53	22	22	22	31	31	31	31
53	53	53	22	22	22	31	31	13	13
52	52	52	23	23	23	13	13	13	13
52	52	52	23	23	23	12	12	12	12
51	51	51	24	24	24	12	12	11	11
51	51	51	24	24	24	11	11	11	11

Fig. 4: A sample Layout obtained from Layout Planning module.

5. CONCLUSIONS

In the present work, a simulation based XY-Oscillatory Sequencing algorithm is developed for Layout Planning of FMS with an objective of minimising the material handling cost. There exists a provision to fix a particular facility at a desirable place as per the requirements. It is possible to leave a particular area of the grid vacant, if required. The algorithm takes the interaction between the different work-stations into account and obtains the best fifteen layouts so that the user can select a layout among them based on the qualitative requirements.

6. ACKNOWLEDGMENT

The authors express their sincere thanks to the Department of Science&Technology, Government of India for the financial support rendered to carryout this work under the sponsored project.

7. REFERENCES

1. Stecke,K.E. "Design, Planning, Scheduling and Control problems of FMS", Proc of the First ORSA/TIMS Special International Conference on Flexible Manufacturing Systems, pp 1-7, Ann Arbor, Aug, 1984.
2. Kouvelis,P. "Design and Planning Problems in Flexible Manufacturing Systems: A critical Review", Journal of Intelligent Manufacturing, V3 no.2, pp 75-99, 1992.
3. Tompkins,J A. and White.J.A., Facilities Planning and Design, John Whiley and Sons, New York, 1984.
4. Francis,R.L. and White.J.A., Facility Layout and Location: An Analytical Approach, Prentice Hall, Englewood Cliffs, NJ, 1974.
5. Sahani,S. and Gonzalez,T.," P-complete approximation problem", Journal of the association of Computing Machinery, V 23, pp. 555-565, 1976.
6. Drezner,Z., "DISCON: A new method for the layout problem", Operational Research, V 28, pp 1375-1384, 1980.
7. Picone,C.J. and W.ilhelm,W.E.. "Perturbaation Scheme to improve Hillier's Solution to the Facilities Problem", Management Science, V 30, pp 1238-1249, 1984.
8. Heragu,S.S. and Kusiak,A., "Machine Layout Problem in Flexible Manufacturing Systems", Operations Research, V 36, pp 258-268, 1988.
9. Bazaraa,M.S., "Computerised Layout Design: A branch and Bound Approach", AIIE Trans, V 7, pp 432-437, 1975.
10. Rosenblatt,M.J., "The dynamics of Plant Layout", Management Science, V 32, pp. 76-86, 1986.
11. Soldberg,J.J and Nof.S.Y , "Analysis of Flow control in alternative manufacturing configurations", Journal of Dynamic Systems, Measurement and Control, 1980.
12. Kouvelis,P. and Kiran,A.S., "The Plant Layout Problem in Automated Manufacturing Systems", Annals of Operations Research. V 26, pp. 397-412, 1990
13. Heragu, Sunderesh.S. and Kusiak, A., "Machine Layout Problem in Flexible Manufacturing Systems", Operations Research, V 36, No. 2, pp 258-268, 1988.
14. Elmaraghy,H.A., "Simulation and Graphical Animation of Advanced Manufacturing Systems", Journal of Manufacturing Systems, V 1, n.1, 1982.
15. Rathmill,K., Greenwood,N. and Houshmand,M., "Computer Simulation of FMSs", Proc. of second international conference on FMS, Brighton, 1982.
16. Carrie,A., Adhami,E., Stephens,A. and Murdochi , "Introducing a Flexible Manufacturing System", International Journal of Production Research, V 22, 1984.
17. Martin,D. and Pritsker Associates , "Simulation in the life cycle of Flexible Manufacturing Systems", Proc. First ORSA/TIMS Special International Conference on Flexible Manufacturing Systems, Ann Arbor, Aug, 1984.
18. Zoller,K. and Adendroff,K., " Layout Planning using Computer Simulation", AIIE Transactions, V 4, no.2, pp 116-125, 1972
19. Subrahmanyam, V.H.R., and Ramesh Babu, N., "An Object Oriented Simulation Approach for Optimal Configuration of FMS", Proc of 17th AIMTDR Conference, Warangal, India, pp 976-700, 1997.

CONSTRAINT SUPPORT FOR ENGINEERING DATABASES

Ka-Lok Lau and Angela Goh
School Of Applied Science
Nanyang Technological University
Nanyang Avenue, Singapore 639798
Tel: (065) 7994929;Fax: (065) 7926559
email: kllau@sentosa.sas.ntu.ac.sg, asesgoh@ntu.edu.sg

ABSTRACT

Many engineering applications such as Computer-Aided Design and Manufacturing (CAD/CAM) are now supported by advanced database models that have been developed specifically to support complex applications. Common to all these applications is the need to maintain consistency of the stored data. This paper describes a constraint manager whose goal is to be extensible, portable and loosely coupled onto various database systems. This system will provide users with the ability to specify constraints on data items. Furthermore, violations of constraints will also be detected and reported.

KEYWORDS

Integrity Constraint Management, Engineering Databases, Database Management

1. INTRODUCTION

Recently, an entire new range of non-traditional applications have emerged, whose demands cannot be met by traditional database systems. A classic example is in Computer-Aided Design/Computer-Aided Manufacturing (CAD/CAM) systems, which require more expressive data modelling capabilities and application models. Objects and manipulations on such objects are more complex. In addition, the concept of data has changed to incorporate multimedia types including text, images, voice, graphics and many other objects. Many researchers have focused on incorporating an object-oriented framework to support complex database applications development. As a result, object-oriented databases, deductive databases and deductive object-oriented databases have emerged. However, there is still a lack of support for constraint management in these systems.

In this paper, constraint management is defined in the context of engineering applications. A constraint management system is described. Its goal is to be extensible, portable and loosely-coupled onto various database systems. This system will provide users with the ability to specify constraints on data items. Violations of constraints will also be detected and reported.

2. BACKGROUND

A major concern in engineering databases is the need to maintain data integrity. Constraints are the rules whereby such data integrity is ensured. Any violation of the constraints would result in invalid data values associated with a data object. As a consequence, violation must be detected and corrective action taken to prevent data inconsistency.

[1-7] lists some of the vast amount of research which has been undertaken into providing facilities to specify, validate and monitor constraints. Surveys of such work can be found in [8,9]. Frameworks have been proposed on the construction of constraint managers[4,6]. However, there has been a lack of reported work on the actual development and evaluation of such systems. Although there are generic systems [10,11] which provide constraint management support, these are tightly coupled with their respective database systems. This implies that in order to take full advantage of the constraint management functions, the particular database system has to be used. If one's application is already based on another database system, conversion to one of these systems would be needed. For those

who have heavily invested in a particular database system, this will not be acceptable. The motivation of the project therefore stems from the need to provide a satisfactory constraint manager and to retrain the use of an existing database system. The proposed solution is an extensible and portable constraint management system which is loosely coupled onto various database systems.

3. OVERVIEW OF THE CONSTRAINT MANAGEMENT SYSTEM

The architecture of the constraint management system is given in Fig. 1. It comprises of three modules which interact with each other as well as with application programs and the database system. The three modules are: the Host Language Interface, the Constraint Evaluation and Execution Module and the Storage Control Layer. The Host Language Interface enables the users to specify constraint within a host language. Such constraints are registered and communicated to the Constraint Evaluation and Execution Module.

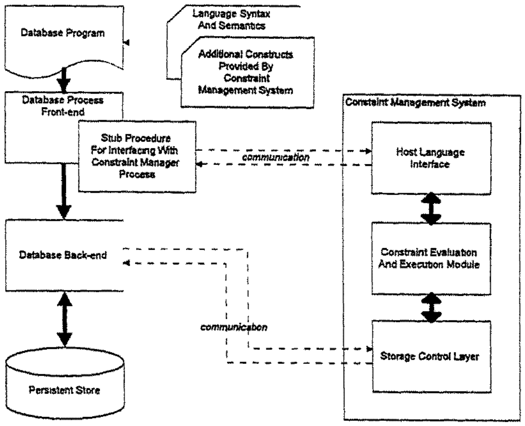


Fig. 1 : The Architecture Of The Constraint Management System

The Constraint Evaluation and Execution Module is the main activity centre of the system. It handles calls from the Host Language Interface and evaluates the constraints based on costs provided by a costing function. Depending on the outcome of the evaluation, certain actions may be executed.

The Storage Control Layer facilitates communication between the other modules of the constraint management system and the underlying database. If the database system is changed, only this aspect of the constraint management system requires modification. Database access is carried out on behalf of the Constraint Evaluation and Execution Module. It also provides statistics and monitoring information.

4. THE HOST LANGUAGE INTERFACE

Various databases were selected to be the underlying platforms. The first is E/Exodus, an object-oriented database systems with a C++-like language. This public domain system is an extensible system which provides sufficient primitive functions to allow further database features to be built. Another platform is CORAL, a deductive database system. A third example is CORAL++, which combines the deductive features of Coral with object-oriented concepts. The different choices were selected to demonstrate that the objectives of extensibility and portability are met.

The Host Language Interface extends each of the underlying database languages by providing constructs thereby providing:

- Rule Declaration: This concerns the establishment of a set of semantics for which constraint rules can be specified. This includes the type of keywords needed to define the start and end of a rule declaration.

- Operators: These include logical operators such as AND, OR and NOT which are used within the construction of rules.
- Action: This specifies the action to be taken when the constraint evaluates to a certain condition. Examples of actions include an immediate abort of transaction, reverting to an original database state, etc.
- Evaluation Priority: In cases where more than one rule applies to a database state, priority is used to determine the order of rule execution.
- Reporting Level: This determines the level of reporting to be carried out when an event occurs and some action is required.

The implementation of the Host Language Interface is depicted in Fig. . It consists of three loosely-coupled units: the language preprocessor, the constraint registration unit and the communication unit.

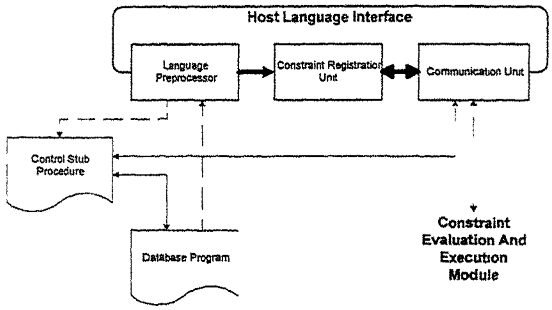


Fig. 2 : The Implementation Of The Host Language Interface

The language preprocessor takes a database program as its input and parses it to extract the set of constraint rules. For each constraint rule encountered the languagė preprocessor will insert a control stub procedure at appropriate locations in the database program. This control stub procedure contains the necessary code, written with the constructs provided within the native language, to communicate with the constraint management system.

The set of constraint rules extracted by the language preprocessor will be passed onto the constraint registration unit to be stored in the constraint management system. In order to do so, the constraint registration unit makes use of a communication unit to pass the constraints set to the constraint evaluation and execution unit.

5. THE CONSTRAINT EVALUATION AND EXECUTION MODULE

The Constraint Evaluation and Execution Module receives all the constraint rules from the Host Language Interface and stores them in its own storage area. Whenever a constraint evaluation call is received from the Host Language Interface, this module will look up the corresponding rule from the stored set, evaluates it and returns the result back to the Host Language Interface.

In Fig. , the implementation of the Constraint Evaluation and Execution Module is shown. It consists of three classes: the *Cost Function*, the *Evaluation Unit* and the *Execution Unit*. The *Cost Function* class computes the cost of evaluating a certain constraint thus helping the constraint management system to decide on how to handle such an event. It uses the data collected at each of the three layers in the constraint management system to calculate the cost. As each database system is different, the cost will also be different. Therefore a change in the database system will require a change in the cost function. Thus encapsulating this cost function as a class makes it much easier to extend or modify.

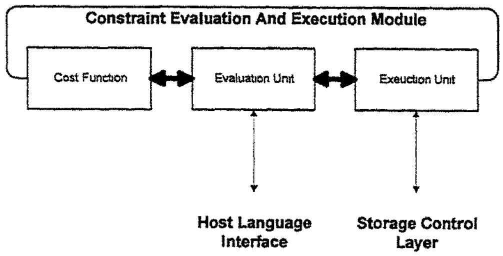


Fig. 3 : The Implementation Of The Constraint Evaluation And Execution Module

The *Evaluation Unit* class handles the incoming calls from the Host Language Interface. In this unit, each constraint that needs to be evaluated is first checked against a set of constraints stored in the constraints registry. Associated with each constraint will be a cost of evaluating that constraint. If the evaluation cost is too high then it passes control back to the host interface for direct evaluation. If the cost is below a threshold, the control is passed to the *Execution Unit* class which will then execute the actions associated with the constraint. The *Execution Unit* communicates with the Storage Control Layer during the execution of the actions required for the particular constraint. These actions are determined in the *Evaluation Unit.*

6. THE STORAGE CONTROL LAYER

The basic function of the Storage Control Layer is to provide a high level generic interface to support communication with an underlying database system. In the context, of this project its use is in providing the constraint evaluation and execution process with a unified interface to the underlying storage structure so that even a change in the underlying structure will not require a re-engineering of the whole constraint management system. This provides the constraint management system with physical data independence.

The storage control layer can be visualised as in Fig. . The *Retrieval Class* handles all the communication with the underlying database as well as the incoming calls from the constraint evaluation and execution unit. These two functions have been merged into one class so as to minimise the overheads involved if it were separated into two distinct classes.

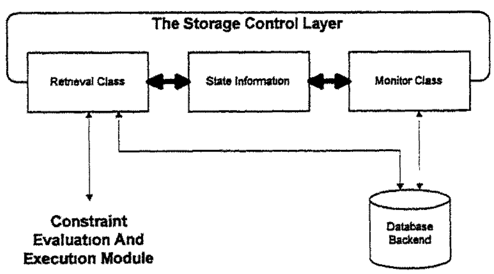


Fig. 4 : The Implementation Of The Storage Control Layer

The *State Information* class contains all the state information necessary such as which data item has been accessed the most and which access is the most time-consuming. In this way data can be gathered to enable a cost function to be derived for use in the constraint evaluation and execution process.

The *Monitor Class* is a class which provides an extra functionality in monitoring the status of the underlying database. This class can also be used to intercept any direct calls from a database program. This can be used in trapping any rogue calls into the database that may cause the underlying database to become inconsistent. For example by deleting data from the database that is being referenced to from other sections in the database.

7. SAMPLE APPLICATION

There are numerous applications which can be used to illustrate the constraints faced in any industrial process. An example selected is the constraints faced by a robotic arm during a manufacturing process. Fig. 5 shows a robotic arm that is required to a processed part from Machine A to Machine B. The time required for each part to be processed on each machine is dependent on the part and the process being carried out. A buffer is used to synchronize the time differences between the machines.

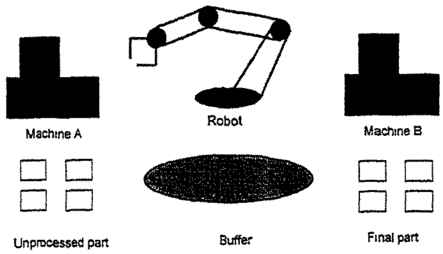


Fig. 5 - A Manufacturing Process Example

In this example, we focus only on the constraints faced by the robotic arm. The arm needs to be notified that Machine A has completed processing a part. The arm will then move this completed part to Machine B. However, if Machine B is currently processing another part, the arm will place A's completed part in the buffer. If the buffer is full, the arm must ensure that Machine A idles until Machine B completes. An example of a C++ class which is used to specify and monitor the robotic arm constraints is given below.

```
class Robotic_Arm {
        /* private and public declarations omitted*/
        constraints:
                rule:   A_Empty
                                if A.current_state = empty
                                then move_part(New_Parts,A)
                                        A.process()
                rule:   A_Finished
                                if A.current_state = finished
                                then
                                        if B_Finished or B_Empty
                                        then move_part(A,B)
                                                B.process()
                                        else not Buffer_Full
                                                move_part(A,Buffer)
                                                A.current_state = empty
                rule:   B_Empty
                                if B.current_state = empty
                                then wait Buffer_Not_Empty or A_Finished
                rule:   B_finished
                                if B.current_state = finished
                                move_part(B,Finished_Parts)
```

```
                                B.current_state = empty
                  rule:     Buffer_Not_Empty
                                if Buffer not empty
                                then move_part(Buffer,B)
                                        Buffer not full
                                        B.process()
                  rule:     Buffer_Full
                                if Buffer = max_number
                                then signal full
    };
```

The above program source is input into the language preprocessor of the Host Language Interface. This generates a list of data dependencies between different objects. This list is then registered into the constraint management system whose task is to monitor the events related to the firing of the rules.

8. CONCLUSION

The paper presents a constraint management system which can be loosely coupled to current database systems. It has been shown that this is particularly suitable for those database systems using compiled-language support. The host language interface enables a constraint manager to be coupled onto different database systems. In fact, the application of such a constraint system is not limited to databases alone, as is illustrated in the sample application.

9. REFERENCES

1. Grefen, P., et al, "Integrity Control In Advanced Database Systems", IEEE Data Engineering Bulletin, Vol. 17, No. 2, pp.9-13, 1994.
2. Asirelli, P., Inverardi, P. and Mustaro, A., "Improving Integrity Constraint Checking In Deductive Databases", 2nd Int. Conf. on Database Theory ICDT'88, pp.72-78, Bruges, Belgium,1988.
3. Grefen, P. and Apers, P., "Parallel Handling Of Integrity Constraints On Fragmented Relations", 2nd IEEE International Symposium On Databases In Parallel And Distributed Systems, pp.138-145, Dublin, Ireland, 1990.
4. Grefen, P., "Combining Theory And Practice In Integrity Control: A Declarative Approach To The Specification Of A Transaction Modification Subsystem", Proceedings Of The 19th VLDB Conference, pp.581-591, Dublin, Ireland, 1993.
5. Seljée, R., "A New Method For Integrity Constraint Checking In Deductive Databases", Data & Knowledge Engineering, Vol. 15, No.1, pp.63-102, 1995.
6. Karadimce, A.P. and Urban, S., "A Framework For Declarative Updates And Constraint Maintenance In Object-Oriented Databases", IEEE 9th International Conference On Data Engineering, pp.391-398, Vienna, Austria, 1993.
7. Moerkotte, G. and Karl, S., "Efficient Consistency Control In Deductive Databases", 2nd Int. Conf. on Database Theory ICDT'88, pp.118-128, Bruges, Belgium,1988..
8. Celma, M., Garcia, C., Mota L. and Decker, H., "Comparing And Synthesizing Integrity Checking Methods For Deductive Databases", IEEE 10th International Conference On Data Engineering, pp.214-222, Houston, Texas, 1994.
9. Grefen, P. and Apers, P., "Integrity Control In Relational Databases - An Overview", Data & Knowledge Engineering, Vol. 10, pp.187-223, 1993.
10. Czejdo, B.D, and Taylor, M., "TANGUY: Integrating Database, Rule-based and Object-Oriented Paradigms", Database Systems For Advanced Applications DASFAA91, pp. 339-346, Tokyo, Japan, 1991.
11. Anwar, E., Maugis, L. & Chakravarthy, S., "A New Perspective On Rule Support For Object-Oriented Databases", Department Of Computer & Information Sciences, University Of Florida, Technical Report UF-CIS-TR-92-042, 1992.

DEVELOPMENT OF A CAD/CAM SYSTEM FOR PROGRESSIVE DIES: THE SYSTEM TECHNOLOGICAL WHEEL FOR DIE STRUCTURE DESIGN

Joshua F. Wang and C. Shi
Gintic Institute of Manufacturing Technology
Nanyang Technological University
71 Nanyang Drive
Singapore 638750

X. Xiao and J. Li
State Key Lab. of Die & Mould Technology
Huazhong Univ. of Science and Technology
Wuhan, Hubei Province
People's Republic of China 430074

ABSTRACT

Reported in this paper are the fundamental technologies developed in implementing a PC based CAD/CAM for progressive dies on the AutoCAD platform. These technologies are necessary for the creation, representation, editing, storing, and querying of various applications, such as features, parts, and structures, etc. in designing a die. It has been proven that these technologies are efficient and powerful, ensuring a state-of-the-art CAD/CAM to be developed.

Keywords: AutoCAD Applications, CAD/CAM for Progressive Dies, Design-by-features, Design-by-structures.

1. INTRODUCTION

Progressive dies for producing sheet metal parts in mass production have been widely applied in the past a few decades. The use of this type of dies results in production automation, higher productivity, higher product quality, lower tooling cost, and more complex part shapes attainable. This, together with the advances in basic sciences and technologies, has contributed to the booming in mechanical, communications, electronics, and light industries world-wide. Performing piercing, slotting, flanging, bending, drawing, cutting off and a number of other processes at a single set-up, a progressive die is generally very complex in structure, and costly in design and manufacture. In contrast to this wide application of the advanced tooling, there is a lack of specially purposed CAD/CAM systems for its effective design and manufacture, specially PC-based cost-effective software for small and medium enterprises. Most of industrial tool engineers still design and manufacture the dies using generic CAD/CAM systems with little or no customisation. Time needed for design is long, and errors are easy to be made. Thus, there is an urgent need for special CAD/CAM systems for this type of dies to increase the productivity and improve the quality of design. It is even desirable today for such systems to possess some intelligence to help make decisions in the complex design situations(1). In past few years, study of CAD/CAM for progressive dies has received great attention(1-4).

In the end of 1993, two research institutes, Gintic Institute of Manufacturing Technology, Singapore, and Huazhong University of Science and Technology, China, lunched an international collaboration research program. One of the major projects in the program was to develop a computer integrated, PC-based CAD/CAM system for progressive dies. By now, a pre-release version of the system has been completed.

During the course of the project, the Singapore project team has been focusing on die structure design and CAM, while the China team on sheet metal part modelling and progressive die planning.

Reported in this paper is the system technological wheel of the module of die design and CAM. In the following sections, the system's functionality will be summarised, the core technologies that are represented by the system's technological wheel will be described and discussed.

2. HG PRODIE CAD/CAM FOR PROGRESSIVE DIES

The software product is named HG ProDie. It consists of four modules, namely,

Module I: Sheet Metal Part Modelling (HG SheetMetal)
Module II: Progressive Die Planning (HG StripLayout);
Module III. Die Structure Design (HG Die); and
Module IV: Wirecut EDM Programming (HG CAM).

All the four modules are developed on the AutoCAD platform and are integrated into one system. The system supports AutoCAD Release 12 and 13, in DOS, Windows 3.1, Windows WorkGroup, Windows 95 and Windows NT.

In module I, a sheet metal part can be created using a feature-based approach. The part can be unfolded automatically to produce a blank which still carries the original semantic information of features. The drawing of the blank is input to Module II, where the plan of the progressive die is made. Respect views and details of the die plan are output for die structure design, which is done in Module III. Drawings of die plates and components involving wirecut machining are output from Module III and input to Module IV, for generating NC programs. Output from Module III are also bill of materials and purchase order for standard components supplied by vendors.

This paper concentrates on functionality and technologies that help the implementation of the 3rd Module, the HG Die Module.

3. THE THIRD MODULE: DIE STRUCTURE DESIGN

The flow chart of the functionality of HG Die is illustrated in Fig 1. Input to the module is a strip layout drawing, and output from the module are assembly drawings, component drawings, purchase order for standard components, and bill of materials. The plate drawings and the drawings of nested irregular punches are input to the CAM module for generating NC programs.

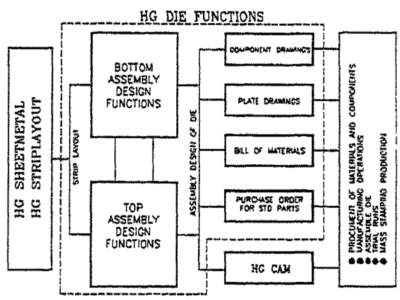


Fig. 1: HG Die Flow chart.

The entire die assembly is divided into two subassemblies, the bottom and the top subassembly, separated from the strip material. The design functions for each subassembly are listed in Fig. 2.

Using the functions for design listed in Fig. 2 is called design-by-structures. The approach of *design-by-structures* is so named because a structure that consists of one or more components, and one or more features on relevant plates, is to be designed in one function, rather than designed separately and sequentially. In using a structure design function, dialogue boxes are opened for the user to search for the key components intelligently from the component library, then, the dimensions of other components and die features are automatically determined. Once the parameters of a structure are determined, its respective drawings are created automatically. In HG Die, a structure is treated as a group of entities, and can therefore be copied, deleted, moved, or modified as needed.

The second approach is known as *design-by-features*. In this approach, individual parts are selected from the library or scratched manually; features on each plates are designed accordingly. This approach is not so efficient as the first one, but contributes to the flexibility of the system. This is because the types of structures and their variations are so many and diverse, it is almost not possible to cover all of them in a CAD system. Wherever a user intends to design a structure whose design function is not available, he can use the design-by-features approach to design any new structures creatively.

To implement the two design approaches in the system, a number of fundamental technologies need to be developed and implemented first. They are presented in the following section.

4. DIE STRUCTURE DESIGN: THE SYSTEM TECHNOLOGICAL WHEEL

The system technological wheel is shown in Fig 3. In the core of the wheel is the AutoCAD System's functions, meaning that all the technologies are built upon AutoCAD. Two layers of technologies are shown in the wheel, with HG Applications technology, HG Algorithms, and HG Engineering Database Systems on the first layer, and HG Features, HG Parts, HG Structures, and HG Standard Component Libraries on the second. The technologies of the second layer are developed from those on the first layer.

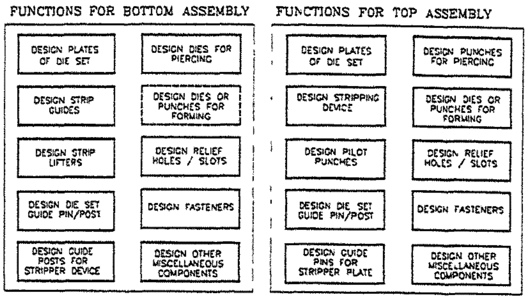


Fig. 2: Structure design functions.

4.1 HG Applications Technology

HG Applications technology is a technology allowing the attributes of any applications to be registered inside of the AutoCAD database itself, implemented using extended entity data. To be consistent, a specification is made on what registering name to use for each application, and what semantic meanings are included to compose the name. Further, a number of library functions are developed to allow the following operations to be made on any particular applications:

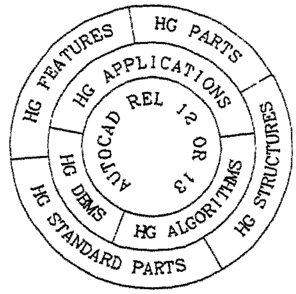


Fig. 3: HG Die System's Technological Wheel.

1. Select an instance of an application. When the user picks up an AutoCAD entity, the function will return one group of entities which together represent one instance of an application.
2. Select an application. All instances of an application are returned.
3. Copy, move, and erase an instance of an application.
4. Remove the semantic meaning (attributes) from an application. This is done by removing the extended attributes data from the AutoCAD database.
5. List all applications registered in the CAD database.
6. Find out the number of instances of an application.

These functions are coded for abstract applications, applicable to any particular applications, such as features, parts, structures, etc.

4.2 HG Algorithms

Algorithms are critical for CAD/CAM systems. They help to develop many automatic design functions. In summary, HG Algorithms include geometric algorithms, such as those for finding intersections between any entities including blocks, finding relative positions of a point to a closed loop (inside, outside, or on it), finding a rectangular area enclosing a set of entities, and shape recognition algorithms. The shape recognition algorithm is used to classify features of piercing holes. This is helpful for automated design of piercing punches and dies. The currently implemented algorithms can recognise 6 basic shapes, together with their dimensions and orientations (see Fig. 4).

Other algorithms that have been developed include those for intelligent searching of standard components from libraries, converting a group of entities into image icons, and numerous list processing functions. All these algorithms become useful tools for the implementation of the CAD system.

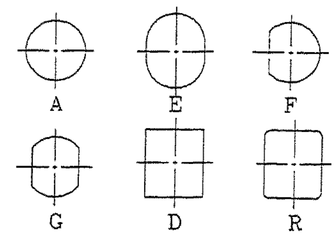


Fig. 4: Hole shapes that can be recognised. Shown below each figure is the shape ID.

4.3 HG Engineering Database Systems

A special database system was implemented in the HG Die module, based on the proposal of Dong (5), for storing and retrieving engineering database tables. In essence, an engineering database table is represented by a lisp function with 3 "keys". We focused generic tables, without considering special cases of repeated range of keys. Simple keys (symbol, string, real or integer number), as well as complex keys (range of values, a range of >= 20 and <= 30 for instance) can be equally treated. A control key is used to allow different values to be returned when more than one values satisfy the retrieval condition.

4.4 HG Features Technology

HG Features Technology include feature categorisation, representation, parameter extraction, query, and their CAD operations like Define, Draw, Copy, Delete and Modify. In HG Features, 5 types of features are defined according to their shapes. They are slot, step, hole, pocket, and chamfer. A *slot* as a through cut made on a side of a plate can be rectangular, circular, or irregular (user-defined). A *step* can be machined on a side or at a corner of a plate. When it is machined on a side, it can be rectangular or circular, or has a user-defined shape. The shape of a corner step is always user-defined. A pocket can be rectangular, with a radius equal or smaller than half of its width, or has a user-defined shape. A chamfer is a feature to be cut at a corner of a plate whose shape is user-defined. Holes are classified into through, blind, and multi-diameter holes, in each sub category, classification is made. For example, a through hole can be with partial thread at one of its ends, and can be tapped. A blind hole can have or have no thread, and can be bored or drilled. Multi-diameter holes can be countersunk or counterbore. Piercing die holes are special multi-diameter holes. Holes with three diameters have also been defined as they are needed in some special applications.

Features that are definable with parameters are created automatically after being defined. For user-defined shapes, the entities are picked up and some necessary processing is made (trimming or changing layer, for example) and the definition is added to the drawing database. Feature's operation methods can be inherited directly from HG Applications methods, and can be extended for their own speciality.

4.5 HG Parts Technology

The Parts Technology is mainly referring to the technologies for registering part attributes to the drawing database. Two categories of parts are considered, namely, standard parts or catalogue parts which are procured from part vendors, and in-house machined parts. For the former, 3 attributes are defined. They are Part Name, Purchase No., and Quantity. For the later, 5 attributes are defined by default: Part Name, Drawing Code, Material, Hardness, and Quantity. A tool is provided to allow the user to define his own attributes without any limitations if the above 5 attributes do not meet the requirements. Furthermore, registered in the CAD database are also the respective views of drawings. The registration of part attributes can be done by calling a library function, and can be performed interactively. Registered data can be removed. A respective view can be deleted, copied. and moved. The attributes change dynamically as these functions are applied. Part information can be queried, and extracted, to produce the purchase order for catalogue parts, or bill of materials.

4.6 HG Structures Technology

A *Structure* is a Group of applications, containing parts (catalogue or in-house machined) and features (parameric or user-defined). A *structure* performs one or more functions in a die during stamping processes. Typical structures are: piercing punches, strip guide lifters, and fastening screws, etc. A structure may be a basic design unit. There are two types of structures, namely,

- Free structures. Their location is not constrained and the number of its instances is not limited.
- Constrained structures. Their locations are determined during strip layout and the quantity has already been determined, too.

The division of the two types of structures is useful as free structures can be *copied* and *moved*, while constrained structures cannot. Example of free structures are stripping springs, stripper blots, fastening screws, strip guide lifters, ejectors, etc. Typical examples of constrained structures are piercing dies or punches, bending dies or punches, etc.

The attributes of a structure include its name, ID or label, view, and reference location. The ID is used for linking a plane view and a side (front, back, left, or right) view. The ID must be unique. The view only indicates that it is in a plane view or in a side view. Manufacturing information is all found in a plane view, while side views are only for presentation purposes.

The following operations are available for a free structure: Copy, Delete, Move, Undefine, Select, Query, and Select, where Undefine is for removing the attributes from the graphics entity, and Select is for selecting all the instances of a structure to be the "last" selected set of entities. The set can then be treated as required (delete, change properties etc).

4.7 HG Standard Parts Technology

The technology of HG Standard parts include the use of HG Engineering Database technology to set up the databases of the catalogue data, retrieval of the databases to fit the design situations, and automatic creation of the drawings. Currently, only Misumi catalogue is considered in HG Die, which is the most widely used catalogue in Asia countries.

5. DISCUSSIONS

In all the technologies presented above, the attributes of these applications are stored in extended entity data. The advantage of the approach is design consistence. This is because the design data is stored in and retrieved from only the AutoCAD drawing database. No external databases are used. The efficiency versus database size was a practical issue in implementing the above technologies. Addition of extended data as attributes of applications would result in increased drawing databases and slowing down the system. This is true specially when all the relevant entities need to carry extended entity data.

need to carry extended entity data. One way to reduce the database size is to create a symbol for an application to carry attributes. It can be placed on a drawing layer frozen to the user so as not influence the design presentation. Some of the applications have been implemented this way. Part is a typical example. Corresponding to each types of parts, a symbol is used to carry the part attributes. This also helps simplify the algorithm for counting the quantities of instances of an application.

In the implementation, the AutoLISP Application Programming Interface (API) was used. In this language, it is very easy to write highly abstract functions. The lower level of functions can always inherit the operations of the higher level ones. In this way, the hierarchy of technologies is constructed. The mechanism is implemented by using a function name as a variable to be input to another function. In applying the function which has function names as input variables, different input function names will lead to perform different operations on different objects. Starting from HG Application Library functions down to a particular feature (through hole for instance) function, this mechanism has been used. In this regard, the hierarchy of the technologies is shown in Fig.5, where further classification of features is not shown, for simplicity.

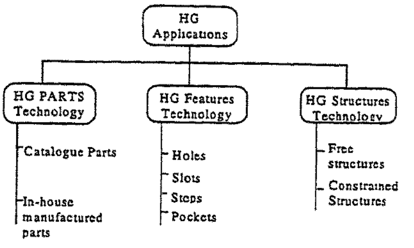


Fig. 5: Hierarchy of HG technologies.

6. CONCLUSIONS

HG ProDie was developed as a third party software of AutoCAD for design and manufacturing of progressive dies. In the die structure design module, two approaches are adopted in design, design-by-structures and design-by-features. The former helps for high productivity and automation, while the latter helps for more flexibility of the system. The system technological wheel has been used to represent the technologies implemented for developing the entire HG Die Module. The key technologies are HG Applications Technology, HG Algorithms, and HG Engineering Database Systems Technology. From these, HG Features Technology, HG Parts Technology, HG Structures Technology, and HG Standard Components Technology have been developed. It has been proven that a successful CAD system is based on appropriate fundamental technologies.

7. ACKNOWLEDGEMENTS

National Science and Technological Board of Singapore and National Natural Science Foundation of the People's Republic of China are acknowledged for their support of this international collaboration project (Project #C93-S-113A).

8. REFERENCES

1. F. Wang, X. Xiao, J. Li, "On development of Intelligent CAD/CAM Systems for Progressive Dies," presented in Int. Conf. on Intelligent Manufactring. '95, Wuhan, China, June 10-13, 1995.
2. F. Wang, & L. Chang, "Determination of Bending Sequence in Progressive Die Design," Eng. Manufacture, Vol. 209, pp. 67-73, 1995.
3. F. Wang, B.Song, & Robert K.L. Gay, "A Feature Based Approach to Modelling Sheet Metal Product Information," Proc. of 2nd Int. Conf. on CIM, Sept. 6-10, Singapore, A. Sen, J. Winsor & R. Gay, Ed., Pub. World Scientific & Global Pub. Services, Vol. 2, pp.22-28, 1993.
4. J. Li, Z. Li, X. Xiao, "A Product Model for Sheet Metal Parts," J. of Huazhong Univ. of Sic. & Tech., Vol. 19, pp. 49-53, 1991.
5. Dong Ren Yang, editor: The Micro Computer CAD Technology: The Development and Applications of AutoLISP Programs, pub. by Chengdu University of Science and Technology, Changdu Institute of Tools and Dies, Chegdu Engine Corp., Chengdu, China, 1983.

OPTIMIZATION IDEOLOGIES AND APPROACHES IN CIMS DEVELOPMENT AND IMPLEMENTATION

M.H. Xiong, Z.Y. Han, Y.F. Zhong and J. Zhou
CAD Center of Huazhong University of Science and Technology
Wuhan 430074 , China

ABSTRACT

CIMS has been considered as the key to improve the productivity, efficiency and profitability of enterprises for many years. But enterprises faced with developing and implementing their CIMS may take the tremendous risk of failure because of the CIMS complexity and little project experience. This paper picks up an important theme about the CIMS Optimization Ideologies(CIMS/OI) and approaches in CIMS development and implementation, which is helpful for enterprises to carry out their CIMS projects. Particularly, the expression of CIMS optimization problems, a new process of CIMS development and implementation and a CIMS optimization approach based on 'coordination — optimization' structure mode are proposed and studied in the paper.

KEYWORDS

Computer-integrated Manufacturing System, Optimization, Development and Implementation, Structure mode

1. INTRODUCTION

CIMS uses computer technology to integrate various components of manufacturing enterprise into a single cohesive system by facilitating prompt and efficient exchange of information between these components. It is considered as 'Factory of the future' models. So CIMS is now the popular theme for most of the manufacturing enterprises in P.R.C, and these enterprises want to increase the benefits and the production flexibility to meet today's changing market demands by developing and implementing this new technology.

Like any manufacturing systems, CIMS is a complex organization of people, machines, materials, procedures and information. Enterprises faced with developing and implementing their CIMS find themselves thrown into environments which are highly complex, with little project implementation experience, and under tremendous pressure to do well right first time. It is this situation that makes some enterprises fail to develop and implement this project, which has influence upon the hope for CIMS in other enterprises.

Over some years, enterprises in China have made substantial investments in CIMS. Some of these systems have gained much publicity for their contribution to productivity. But while some have won honor for their CIMS-DI, others have failed to meet their expectations about benefit and substantial performance of CIMS in enterprises. An estimated more than 50 ~ 75% of projects fail to meet these expectations[1]. So CIMS-DI has been considered as a 'two-edged sword'[2] by some people.

Clearly, the key ideologies and methodology in CIMS-DI which are not the same as in normal project process must be pointed out and carried out emphatically. The important parts of them are CIMS/OI and their approaches. It's important, efficient and helpful to apply the CIMS/OI to the whole process of CIMS-DI, and this can help the CIMS enterprises to meet their expectations.

2. CIMS OPTIMIZATION PROBLEM

So far, it has been widely approved by personnel all over the world that the aim to develop and implement CIMS in a enterprise is to improve its productivity, efficiency, and profitability. In other words, CIMS-DI in a enterprise is applied to optimize the enterprise itself by means of the CIM theory. That includes the core of CIMS/OI.

2.1 Discussion

Obviously, CIMS optimization objectives may be more than personnel can imagine. But generally, this objectives can be discussed through analyzing the concept model of CIMS optimization problem(CIMS-OP).

A CIMS-OP Copt can be expressed by a 5-tuple as below.

$$Copt = < X, Po, O, K, Mo >$$

where :

X describes state space of a CIMS-OP.

Po describes optimization principles or objective set, e.g. T, Q, C, S in general.

O describes optimization objectives and their relation set.

K describes knowledge set about CIMS optimization objectives and their relationship, and it includes CIM ideology, CIMS reference models, and CIMS architecture.

Mo describes CIMS optimization approaches, techniques, and tools.

Actually, CIMS optimization can be determined by the transformation from state space (X) to optimization space (X^*), which, in other words, can be also defined as from enterprise As-Is model to To-Be model. So it is very important to understand and distinguish correctly and comprehensively the nature and regularity of CIMS optimization.

2.2 CIMS Optimization Objectives and Their Relationships

Through analyzing the nature and aims of process of CIMS-DI, we can define CIMS optimization objectives and their relation set (O) as a 3-tuple:

$$O = < BE, CS, CD >$$

where :

BE describes business entity and relations around it, and it can be further expressed as a 3-tuple:

$$BE = < Bp, Pp, Os >$$

Of above three tuple variables, Bp describes enterprise management procedure and relations around it, which is a description of a business entity from its internal construction. Pp describes enterprise production procedure and relations around it, and it can be expressed by product development procedure and relations around it, product structure and relations around it, and manufacture procedure and relations around it. Os describes enterprise organization structure and relations around it, and it represents business entity from some main respects such as enterprise personnel structure, organization setup and so on.

CS describes a CIMS itself of a enterprise which can be expressed by CIMS universal set of its components and all relationships between them.

CD describes a process of a CIMS-DI which can also be further described by means of state of a CIMS-DI, CIM ideology instructing how to work with CIMS, approach and tool set of how to develop and implement CIMS, and the set of personnel involved in CIMS-DI.

Undoubtedly, CIMS-DI is not the process of computerizing the current operation activities of a enterprise. In order to get profitability, the process of CIMS-DI has to develop some new approaches, tools and methodologies which must include and carry out the CIMS /OI. The contents discussed above is great helpful to understand and analyze CIMS/OI.

3. CIMS DEVELOPMENT AND IMPLEMENTATION

In general, a reasonable and optimum process of CIMS-DI must make a CIMS enterprise to be assured of achieving the best benefit/investment. So, it is very important to analyze and develop a optimum process of CIMS-DI.

CIMS-DI is a process which combines 'Top-down' and 'Bottom-up', and can be generally divided into several phases such as feasibility study, preliminary design, detailed design, system implementation, and system operation and maintenance. This division has a strict structure but little flexibility because essentially it's a structured life-cycle development approach.

Because it is quit difficult for a enterprise to determine completely and correctly the enterprise demands and the CIMS functions in the feasibility study phase or system design phase, the modifying and repeating work is necessary in CIMS-DI as the enterprise management objective and environment has been changed, and this modifying and repeating work is helpful for the enterprise to develop a CIMS of good quality and achieve the best benefit/investment.

Therefore, a new process of CIMS-DI is proposed in this paper, which is mainly divided into such phases as CIMS master plan, implementation sequence determination, detailed design and system implementation. Their relationships between the main phases are illustrated in figure 1, and the condition of input, output and environment(control and mechanism) in every phases of CIMS-DI are showed in figure 2 by means of the structured design tool IDEFo.

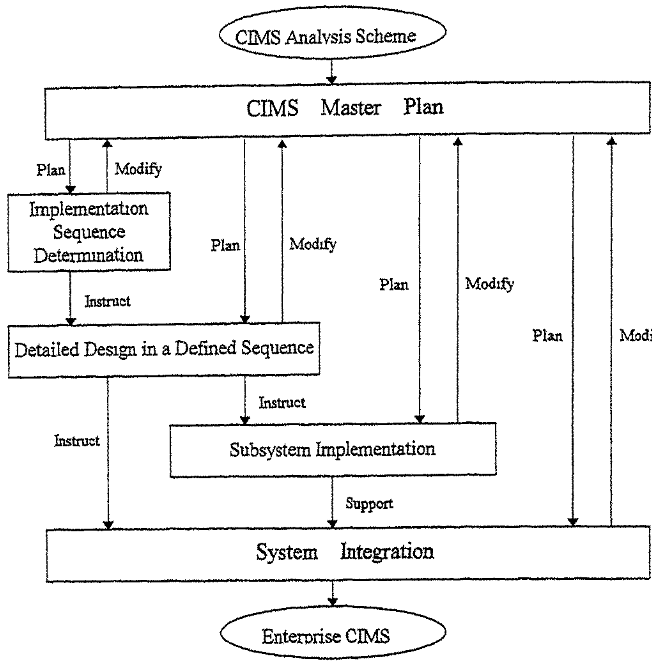


Fig. 1 : Relationships between main phases of CIMS-DI

Compared the above process with the usual development process, the difference is that this process breaks the strict linear development mode and there is the feedback from the follow-up processes of CIMS master plan, and according to this feedback the master plan is modified.

The system detailed design, implementation and integration are completed in accordance with determined implementation sequence, and so, having been meeting the demands of follow-up processes, the amounts of design work are decreased and some repeated and useless work are avoided. Therefore, this development and implementation process is a ‘process for reality’ or a ‘process for application’.

4. CIMS OPTIMIZATION APPROACH

The CIMS optimization approach in CIMS-DI is not, as yet, well defined and studied because of its wide-range techniques, complex relationships between subsystems, especially the key factor of ‘personnel’ involved in CIMS.

Actually, the structured mode of CIMS optimization can be proposed generally if the study attention has centered on the nature of CIMS and CIMS-DI. In this paper, a CIMS optimization development approach based on ‘coordination — optimization’ structure mode is put forward for the process of CIMS-DI, and it can be illustrated in figure 3.

The ‘Coordination — optimization’ structure mode is essentially a optimization development model with two-layer structure corresponding to the decomposition problem for CIMS development. Its important components are coordination model (CM), subsystem optimization model (OM) and explainabler.

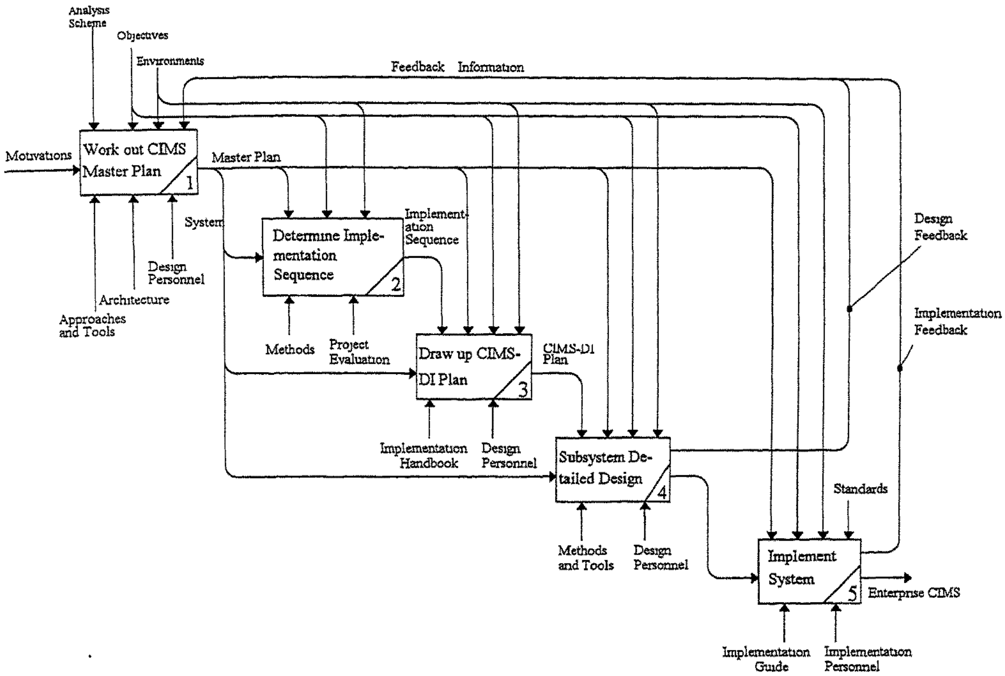


Fig. 2 : A new process of CIMS-DI

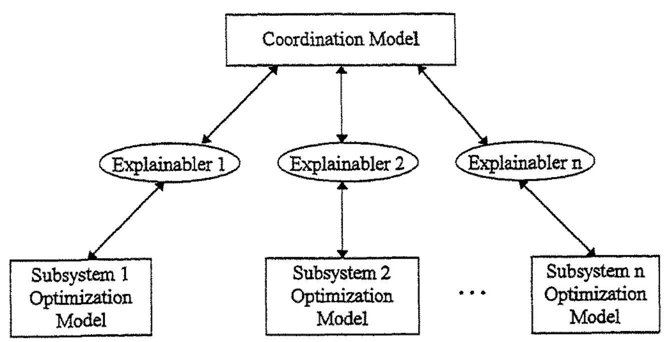


Fig. 3 : 'Coordination — optimization' structure mode of CIMS optimization development

Coordination model (CM) :

CM is described by coordination principles, decision variable set made by CM, coordination variable set through explainabler and constraints over coordination procedure. CM considers problems such as resource and time constraint, total demands for relationships between all subsystems and integration. It works on the optimization plans of each OM, and gives assurance that the whole system can be integrated and the total optimization objectives can be achieved.

Subsystem optimization model (OM) :

OM is determined by optimization principles, decision variable set passed through explainabler, coordination variables made by every OM, constraints over OM, reference model of OM and optimization plan of OM. It works by means of rebuilding and optimizing the drawn structure from the reference model.

Explainabler :

Explainabler links CM and OM, and has the explanation function which explains the decision variables from CM and the coordination variables from OM.

Applying the 'Coordination — optimization' structure mode to the process of CIMS optimization development can not only make the developed subsystems achieve and approach the optimum objectives through optimizing the decomposed subsystems separately, but also control and coordinate the relations of each development problems.

The 'Coordination — optimization' structure mode describes the form and process of CIMS optimization development. In reality, this model is difficult to be operated. For simplifying this process, by consulting the resolving approaches for usual two-layer decision problems, a resolving scheme for the 'Coordination — optimization' structure mode can then be proposed in a relatively simple way (which are not discussed in this paper) .

5. CONCLUSIONS

Compared with usual engineering projects, it is more difficult, costly and resource-intensive for enterprises to develop and implement their CIMS. In the development process, the instructions of the correct ideologies and approaches are required.

The CIMS Optimization Ideologies(CIMS/OI) is very important for CIMS development and implementation . In this paper, the concept model of CIMS optimization problem has been expressed and analyzed in order for CIMS/OI to be understood correctly and comprehensively, in particular CIMS optimization objectives and their relation set have been described.

A new process of CIMS development and implementation has been presented, in which the main phases are CIMS master plan, implementation sequence determination, detailed design and system implementation. The difference between this process and the usual one is that in this process the strict linear development mode has been broken and there has been the feedback from the follow-up processes of CIMS master plan. This new process of CIMS-DI is very useful for enterprises to draw up their development plan.

Generally, a CIMS optimization development approach based o 'coordination — optimization' structure mode has been proposed which tries to coax CIMS developers to program their framework for CIMS.

The CIMS enterprises have the chance of achieving the best benefit/investment, and this can be made possible by the application of CIMS/OI principles and its approaches. This paper has only planned CIMS/OI and its approaches, thus the research work ought to be continued in the future.

6. REFERENCES

1. D.O'Sullivan, "Development of integrated manufacturing systems", Computer-Integrated Manufacturing System, 5(1),39-53,1992
2. Ebers,M and Lieb, M, "Computer integrated manufacturing as a two-edged sword", Int.J.Operations & Production Manage, 9(2),69-92,1989
3. R.J.Borden, "Organizing for CIM: project management, technology and integration", Computer-Integrated Manufacturing System, 14(2),60-69,1991
4. Y.K.Son,C.S.Park, "Economic Measure of Productivity, Quality and Flexibility in Advanced Manufacturing Systems", J.Manufacturing Systems, 6(3),193-207,1990
5. J.G.Demel,R.G.Askin, "A Multiple-Objective Decision Model for the Evaluation of Advanced Manufacturing System Technologies", 11(3),179-194,1992
6. L.M.Camarinha-Mats,H.Pinheiro-Pita,R.Rabelo and J.Barata, "Towards a taxonomy of CIM activities", Int.J Computer Integrated Manufacturing, 8(3),160-176,1995
7. C.A.Snyder,J.F.Cox, "Developing computer integrated manufacturing: major issues and problem areas", Costs and Economics, 17,197-204,1989
8. C.A.Voss, "The managerial challenges of integrated manufacturing", Int.J.Operations and Production Manage, 9(5),33-38,1989
9. Zehenovic,M.Dragutin, "Methodology for design of effective computer integrated manufacturing systems", Robotics and Computer-integrated Manufacturing, 7(3/4),279-290,1990

MANUFACTURING AND PRODUCT DEVELOPMENT IN THE USA, IN PARTICULAR IN THE LOS ANGELES BASIN AND THE STRATEGIC TRAINING AND IMPLEMENTATION OF HIGH MANUFACTURING TECHNOLOGY

Karl-H. Grote
Department of Mechnical Engineering / Engineering Design
Otto-Von-Guericke University, Magdeburg
Germany
Michael L. Walo, Principal Engineer/Scientist
McDonnell Douglas Aerospace, Huntington Beach, California
USA
Jeffrey L. Miller, Sr. Engineer/Scientist
McDonnell Douglas Aerospace, Long Beach, California
USA

ABSTRACT

In order to compete and win on the manufacturing battlefield, many companies have adopted various rapid product development methodologies. In the forefront are Concurrent Engineering (CE), Simultaneous Engineering (SE), and/or Integrated Product and Process Development (IPPD). These approaches share the common objective of performing many of the product development functions early in the product definition phase, in parallel, through the use of multi-discipline teams. In recent years, the development of new technologies including SFM enables these teams to physically realize their concepts at the earliest stages of product definition with significantly reduced risks to the program and associated costs. The successful implementation of these new technologies is dependent on achieving a thorough understanding of the capabilities and benefits of these processes which is accomplished through an educational strategy /1,2/.

KEYWORDS

Product Development, Manufacturing Technologies, Strategic Training, Rapid Prototyping

1. INTRODUCTION

The first step in the educational strategy is to clearly define the product development cycle specifically identifying the requirements of each phase. The available data produced in each phase will be the driver for the Solid Freeform Manufacturing (SFM) processes. The second step is to identify at which phases in the product development cycle, application of SFM technologies will provide the greatest benefit to the Integrated Product Team (IPT) /3/. The third step in the strategy is to provide the process capabilities of available SFM technologies. An understanding of the process capabilities is essential for specifying the appropriate SFM technology for the task. The fourth step is to define the data and preparation requirements for using a SFM process. This step is essential to ensure that the parts to built comply with the capabilities of the SFM process.

2. GENERIC PRODUCT DEVELOPMENT CYCLE

Most Department of Defense (DoD) and commercial operations go through basically the same phases to define, produce, deliver, and support their products and/or systems:

- A study is done to determine need for the product /system
- All candidates are evaluated and the best choice(s) are selected for development
- Testing and fine tuning of the product(s)/system(s) are done
- Product(s)/system(s) are manufactured and delivered to customers
- Product(s)/system(s) are used by customers and supported in the field

2.1 DoD Acquisition Cycle

The five phases of a program, as defined in DoD 5000.2 - Defense Acquisition Management Policies and Procedures, are summarized below:

- Concept Exploration and Definition (CD) - The initial program phase in which the most promising system concepts are defined, alternatives are explored, risks are identified and analyzed, mitigation approaches are developed, trade-off studies are performed, and the optimum concept is selected. Acquisition strategies are planned.
- Demonstration and Validation (DEM/VAL) - The program phase at which major system alternatives are identified and analyzed. Competitive demonstrations are also held to prove and validate concepts defined during the prior CD phase.
- Engineering and Manufacturing Development (EMD) - The program phase where the most promising design approach transitions from development to production. Manufacturing/Production processes become standardized to achieve the most stable, producible, and cost effective product/system design.
- Production and Deployment - The program phase where a stable, efficient production and support base is established. It is also where the system hardware is manufactured, tested, inspected, accepted, and delivered to the customer. Supporting software is developed, verified, and delivered.
- Operational Support - The program phase during which the delivered product/system must be maintained. User training and depot support is required. Any short comings or deficiencies that must be corrected to improve performance are also identified.

2.2 NASA Program Life Cycle

National Aeronautics and Space Administration (NASA) programs use slightly different names and timing for their development phases. Fundamentally, the development process is the same.

2.3 McDonnell Douglas (MDA) Integrated Product and Process Development Cycle

The six phases of Integrated Product and Process Definition are executed during the first three DoD acquisition phases. The contractual acquisition phase requirements will tailor the output of the definition phases for each program. Each acquisition phase will satisfy certain milestone requirements before contracts are let for the subsequent phase.

Configuration Synthesis, consisting of High Level Requirements, Initial Concepts, and Configuration Baseline definition phases, is executed during the Concept Exploration and Definition (CD) Phase. The Conceptual Layout definition phase of Configuration Synthesis will occur during the Demonstration and Validation (DEM/VAL) phase.

The Assembly Layout and Build-To, Buy-To, and Support-To Package definition phases, for product and process development, are accomplished during the Engineering and Manufacturing Development (EMD) phase.

Some acquisition contracts require prototypes for evaluation during the DEM/VAL phase. MDA has successfully used SFM technologies to create prototypes in this phase for several programs. This requires the early development of prototype Assembly Layouts, Build-To, Buy-To, and Support-To Packages, and CAD models for SFM processes. These early packages are refined during the testing and evaluation performed and result in more robust packages when the design is selected for advancement to EMD.

The incremental release of the packages allows the pre-production build efforts to begin. A first article test is performed to validate the processes and define required changes to satisfy production requirements. Throughout this process time, the Build-To and Buy-To Packages are verified to finalize the production processes. The IPT integrates and coordinates the required changes in the processes and updates the packages to a production configuration.

When the Pre-Production Build is complete, the Low-Rate Initial Production (LRIP) phase begins. The Low-Rate Initial Production phase verifies the planning and tooling for efficient serialized production as the planned production rate. During this phase, as changes to the process are identified, the IPT will continue to make alterations to refine the product and process definition. Finally, with the commencement of Full-Rate Production, the IPT retains the responsibility to integrate product and process improvements as the product/system evolves.

It has been widely documented in numerous manufacturing studies from automotives to consumer electronics that the overwhelming amount of recurring costs, or costs that are paid every time a product is produced, are "locked-in" during the first 20% of the product development cycle. This statistic relates to the activity which occurs during concept exploration, analysis, and definition. Once the design concepts and processes have been defined, it is very difficult and seldom cost effective to re-engineer the design.

3. IPPD TRAINING STRATEGIES

The formal IPPD Training Program at MDA, as in other similar companies, is only one portion of an overall strategy to increase awareness of new technologies and share successes among IPT members on all programs and among all components /4,5/.

This training strategy begins with the employees/IPT members gaining an awareness of what is happening in the world of product development. It continues by offering focused training that can be applied by all individuals at any phase of any program. The strategy completes the training cycle by employing successful IPT leaders/members to share successes and also help new IPT's get started.

McDonnell Douglas Aerospace believes that the best approach for the development, delivery and maintenance of a IPPD training program is to use the combined talents of its professional and technical employees. This mix of theoretical principles supplied by our Development and Training group combined with real world experiences supplied by our resident IPPD activists provides students with the best of both worlds. This goal requires, that training programs must be structured, standardized and monitored for functionality in "real-world" applications by employing enthusiastic and successful IPPD activists as course developers and instructors.. Maximum benefit is also achieved by training Integrated Product Teams (IPT's) as a unit prior to implementing IPPD on their respective projects.

Major emphasis must be placed on IPPD concepts, principles, and tools (both systems tools and quality technologies) as well as on awareness of the cultural changes required. Currently, training is targeted at three different but interrelated audiences:

- IPPD Integrated Product Teams
- IPT Team Leaders
- IPT Program Management

A major step toward implementing new technologies is to first thoroughly train the employees that will be participating on Integrated Product Teams (IPT's) in the principles of IPPD. A proven training program has been used at McDonnell Douglas Aerospace for the last four years and has successfully trained over 2,500 IPT members. IPPD IPT training focuses on IPPD principles and quality technologies and includes a variety of simulation exercises and examples. Included in this training is a module on Facilitating Tools which has been expanded to emphasize Rapid Prototyping. This will be discussed in a later section. IPPD Team Leader Training places an additional focus on team building skills and identification of appropriate team membership at appropriate program phases. Program Management Awareness takes the form of a course overview which focuses on management's role in IPPD and the need for their "top down" buy-in for IPPD's success. It also explains basic organizational philosophy and success stories which help educate them on the benefits of the concept. It explains what's in it for them. The program also includes a video presentation that can be used as an introduction for training participants as well as for new hire orientation. The IPPD training course consists of eight separate modules, which are explained in the following paragraphs.

3.1 IPPD Training Module Objectives

Upon completion of the introductory module (Module 1), the students understand basic IPPD practices and how they differ from "traditional" design practices. The introduction sets the stage for all other training modules and gives the students an opportunity to ask questions and clarify any uncertainties they may have.

3.2 IPPD Teams (Module 2)

Upon completion of the teams module, the students will have gained an appreciation of how a typical multidiscipline IPT is selected, how effective communication must be fostered, and how the team may evolve through time. The module avoids a "cookbook" approach, as the composition and structure of each IPT varies with the size and scope of the individual project. Nonetheless, the team will gain an understanding of what is expected of them, the team leader, and their management.

3.3 Robust Design Principles (Module 3)

Upon completion of this module, the students will be able to identify and give examples of the seven Robust Design Principles:

- Design for competition
- Design for customer needs
- Design for manufacturability/assembly
- Design for test and inspection
- Design for reliability
- Design for maintainability/supportability
- Design for product evolution

3.4 Quality Technologies (Module 4)

Upon completion of this section, each IPT member will understand the potential uses of quality technologies in an IPPD environment:

- Quality Function Deployment (QFD)
- Design of Experiments (DOE)
- Statistical Process Control (SPC)

After the Quality Technologies Module, the students begin their simulation exercise which continues to run through lunch on the second day. The students participate in a role playing exercise designed to simulate a typical IPT environment and illustrate many of the IPPD concepts taught to this point. The class instructor plays the role of the customer, while the students interact as suppliers, manufacturing, design, quality, other "illities," and production engineering personnel. In the exercise, students assess the voice of the customer (VOC), develop and critique the design, build a prototype, test, and manufacture a sample product. Throughout this exercise, the instructor varies the conditions of the simulation by issuing customer driven changes and schedule revisions. Upon completion of the exercise, students compile a list of major lessons learned that they can later apply to "real" IPPD activities.

3.5 Facilitating Systems and Tools (Module 5)

Upon completion of this module, students will be aware of the systems and tools available to support their IPT's. They will understand the effect of the IPPD approach on the use and implementation of these tools.

3.5.1 Original Focus of Module 5

This module originally focused on the systems and tools that can enhance the IPT work environment (product, process, support). Product tools, which assist in designing the product and communicating design intent to the factory floor, include computer-aided design and computer-aided engineering (CAD/CAE), as well as simulation and analysis tools. Process tools, which assist in production and delivery of the product, include computer-aided manufacturing (CAM), computer-integrated manufacturing (CIM), process planning, discrete event simulation, and manufacturing resource planning (MRPII). Support tools, including electronic communication and office automation, are those which have an indirect effect on the product and process. Students are provided with a conceptual overview of the key systems tools and how they can assist the product development team. Concepts such as group technology and artificial intelligence are also discussed, as well as data and communication standards. The effect of the computer-aided acquisition and logistics support (CALS) initiative on IPPD is also addressed. At the conclusion of this section, students are given a catalog that summarizes and evaluates specific available software and hardware used for each tool category. The catalog also indicates where in the product life cycle the tool can be applied and whom to contact for information or support.

3.5.2 Modifications to Module 5.

Updates and modifications to the Facilitating Systems and Tools Module are extremely critical to maintain the integrity and credibility of the course. Rapid Prototyping is an example of a technology that requires frequent updates in the Facilitating Tools Module, as it is an emerging technology that is continuously being expanded and improved. Many of our students come from the worlds of design and analysis where they use state of the art technology in their jobs daily. Imagine how the training program would appear to them if we used examples of electronic tools that were out of date or did not truly represent what was being used today? The same holds true for our students from the worlds of manufacturing and operations. Our training must accurately represent the latest advances and lessons learned (both pro and con) in automated planning systems, Computer Aided Manufacturing (CAM) and production control systems. The most important element of this module, however, is how it brings an awareness of the power of new technology to all students /6/.

4. CURRENT TRAINING ON RAPID PROTOTYPING

The MDA training module on facilitating tools was recently expanded to include extended discussion and examples of rapid prototyping technologies. Initially, the students, current or future IPT members, are exposed to the concept of Rapid Prototyping.

The prototyping parts shown to the students have been carefully selected to provide maximum clarification of how this process can be used in the product development process. A specific example showing a part built for an engineering structure is presented to the class. The second example demonstrates how SFM parts may actually be used for fit checks of working prototypes. This has been done where it has been proven more cost effective to build a prototype model instead of making a part out of metal. This is demonstrated by assemblies built for several industries. The students can view assembled housings and components built using RP-Technologies. The third case involves discussion of using various Rapid-Prototyping proceedures to produce metal castings. This was used recently to reduce fabrication time of critical components for several development programs. To complete the training on Rapid Prototyping, the students are asked to provide examples of how SFM technologies can be used on their programs and in their companies. Specific parts are described and the viability of using SFM processes to fabricate those parts for a prototype application is discussed.

Enhancements to Rapid Prototyping training will include a discussion of specific process capabilities of different SFM technologies. This is critical for design engineers who need to make choices on which processes to use. Different SFM processes will be described, and their respective capabilities compared, with respect to tolerances and process advantages for specific features. The primary processes which will be discussed in the IPPD training include:

Σ SLA Stereolithography (SLA)
Σ Selective Laser Sintering (SLS)
Σ Solid Ground Curing (SGC)
Σ Laminated Object Manufacturing (LOM)
Σ Fused Deposition Modeling (FDM)
Σ Direct Shell Production Casting (DSPC)
Σ Ballistic Particle Manufacturing (BPM)
Σ Cross Sectional Prototyping (CSP)

For example, a matrix is currently in development, which is designed to illustrate basic process capabilities of the processes identified above. In addition, SFM capabilities will be compared to conventional process capabilities for processes such as machining, sheet metal forming, and casting. Additional training outside of the scope of the IPD basic program will include modeling techniques and specific process details to help the designers generate CAD models that can be directly used for SFM processes. This is currently being done on an informal basis depending on program requirements.

5. INTEGRATED SCHEDULES

After completing Module 6, students will be aware of the scheduler's role on the team and the effect of parallel activities on schedules. They will understand how to interpret and use a IPPD schedule.

5.1 IPPD Measures (Module 7)

Upon completion of this module, the students will be able to use the IPPD team benchmarking matrix to measure team progress at regular intervals. The IPT members will also be aware of the array of effectiveness measures (quality, efficiency, timeliness, cycle time) that can be monitored to see how IPT's improve overall company performance.

5.2 Next Steps (Module 8)

After completing the final module, students will be aware of some of the initial challenges facing a IPT and how to overcome them. They have the opportunity to ask questions of the module presenters who are successful IPT leaders and IPPD advocates. An other technology, 3D- Digitizing is beeing introduced in the described development process, which will facilitate the more more important RE- Engineering process. The software and the hardware for this technology has no real standard yet - but our research activities go into that direction - also to include 3D-Scanning in a closed development loop with 3D- Modelling and Rapid Prototyping proceedures.

6. CONCLUSION

An educational training strategy is essential for successful product development technologies /7/. A proven training program has been expanded at McDonnell Douglas to include specific details of SFM processes. This has led to increased usage of SFM technologies on several programs. McDonnell Douglas will continue to improve its training programs so that IPT members will be fully capable of effective implementation of SFM technologies to remain on the cutting edge of rapid product development.

7. REFERENCES

1. Miller, J. L., Automating Producibility Analysis for Rapid Product Development. M.S. Thesis. Department of Mechanical Engineering, California State University, Long Beach, 1995.
2. Walo, M., Miller, J. L and Grote, K.-H., "Strategic Training And Implementation Of Solid Freeform Manufactruing Technologies For Product Development", Proceedings of Intelligent Production Systems: Solid Freeform Manufacturing, 3rd International Conference, Dresden, Germany, 1995 also published as MDC - Paper 95HO752, 1995.
3. Miller, J. L. and K.-H. Grote, "SFM Technologies in Theory and Practice in the USA", Proceedings of Intelligent Production Systems: Solid Freeform Manufacturing, 2nd International Conference, 39-46, GFaI Sachsen, Dresden, Germany, 1994.
4. Walo, M. L., A Forecast of Skill Requirements for the Aerospace Workforce in the Metal Fabrication Trades in Southern California in the Next Twenty Years. M.A. Thesis. Department of Occupational Studies, California State University, Long Beach, 1995.
5. Walo, M. L., "Integrated Product Development Training Strategy and Options", McDonnell Douglas Aerospace (MDA) - Operations West Horizon Publication. Vol. 7, #1,1994.
6. Walo, M. L., Mosard, G. R., and Moses, S. E., "Development and Implementation of a Formal Training Program for Concurrent Engineering in an Aerospace Environment", American Institute of Aeronautics and Astronautics (AIAA) Baltimore, MD, 1991.
7. Pahl, G. and Grote, K.-H., "Interdisciplinary Design - Knowledge and Ability Needed", Interdisciplinary Science Reviews,VOL.21, NO.4, 292-303, 1996.

THE KEY FOR RAPID PROTOTYPING TECHNOLOGY: NEW MATERIAL DEVELOPMENT

Y.G.Wang and Leslie K.C.Lim
KINERGY PTE LTD
Singapore

ABSTRACT

Although there are so many superiority for rapid prototyping process than conventional processes, especially the reduction of product cost and time to market, the dimension accuracy and some performances of prototype are still not perfect. At present the most obvious weakness for rapid prototyping technology is that the most materials used are not the final product ones, and there are distortion and shrinkage for prototype in process. The trouble as mentioned above results in the gap between the performance of prototype and requirement of final product. This paper describes the main advantage of sheet-based materials for rapid prototyping, effective way to improve the performance of the materials, and points out the further direction for researching and developing materials.

KEYWORDS

Sheet-based material, additive, distortion, shrinkage

1. INTRODUCTION

Although the appearance of rapid prototyping technology was less then ten years, it has been fast approved and applied by industries due to its superiority which can reduce the time to market and cost , improve quality for products. Now there are so many different kids of rapid prototyping systems (RPS) , and their development was mainly around the improvement of materials used and relevant laminated manufacturing methods . At the beginning the photo-sensitive resin in liquid state was used, and the resin was cured by UV laser in rapid prototyping process. This method (SLA) started the milestone of rapid prototyping technology, but its limitation was exposed along with its application. Than LOM sticking and cutting paper coated by adhesive , SLS selectively sintering powders, FDM melting plastic wire and TDP sticking powder etc. appeared.

Up to now the forming accuracy and speed of rapid prototyping are not perfect. Usually, its dimension accuracy is only +/- 0.1 mm to 0.2 mm. And it is specially difficult to achieve high accuracy in Z direction by reason of distortion, shrinkage and inconsistency of layer thickness along the cross section area . Therefore the post treatment (polishing etc.) for prototype will take up a lot of time. The forming speed of rapid prototyping is quite fast compared with the conventional manufacturing methods, but customers still feel it is not enough, such as the time for making a big and more complicated product needs several days. And most materials used in rapid prototyping are not the final product ones caused by the limitation of laser power and distortion in the processing.

The authors think the development of materials for rapid prototyping is the most important. Its main direction should be as follows:

(a) The materials should be kept best in their solid states during the processing to reduce the irregular deformation caused by the change of state of materials.

(b) The better bases of materials are paper ribbon, plastics ribbon, metal sheets, wire, powder with some additives and special chemical treatments to improve the performance of materials and make the prototypes close to the final product ones.

(c) The development of surface treatment technology, especially metal spraying/castung at room temperature,should be emphasized to produce molds/ dies/ tools.

(d) The professional research centers and companies for producing materials should be set up to save time and cost for the manufactures of rapid prototyping systems and promote the development of materials.

2. THE REQUIREMENT FOR MATERIALS AND ITS IMPLEMENTATION

According to the usage of rapid prototyping parts, there are different requirements for the materials used in RPS as follows :

- ◆ Enough strength: tensile and compression strength.
- ◆ Enough hardness and wear ability.
- ◆ Good finish ability.
- ◆ High operation temperature.
- ◆ High moisture resistance.
- ◆ Small expansion coefficient and shrinkage.
- ◆ High heat conductivity.
- ◆ High electric conductivity.
- ◆ Good elasticity.
- ◆ Good transparency.
- ◆ Good excess material separableness.

For sheet-based materials the requirements as above can be achieved by the ways as follows:

(a) select the suitable base-materials, such as different papers, polymer films, metal foils, and ceramic film etc.

(b) select the suitable adhesives.

(c) add some additives to improve the performance of materials.

So the research and development of materials is the key for rapid prototyoping technology.

3. THE MAIN ADVANTAGE OF SHEET-BASED MATERIALS

The sheet-based materials with adhesive and additives have remarkable advantage to the performance of prototypes as follows:

◆ There is no state change for base-materials in rapid prototyping process rather than obvious chemical and physical changes for liquid resin in SLA process, and physical change for plastics powder/ wire in SLS/FDM process. The chemical and physical changes in rapid prototyping processes will cause the irregular deformation, such as distortion and shrinkage of prototypes which are the main trouble for their dimension accuracy. Therefore the dimension compensation of prototype have to be considered before SLA/SLS/FDM process. The consideration will spend time and result in difficulty specially for the new operators.

◆ The melting temperature of adhesive is much lower then the plastic deformation temperature of base-materials as found in the curing/sintering/melting for the SLA/SLS/FDM processes. So there is lower shrinkage and higher dimension accuracy specially in Z direction in the rapid prototyping process using sheet-based materials.

◆ It is only necessary to cut the contours and some crosshatches of every layer of prototype rather than cure/sinter/melt the whole area of every layer. Therefore the required power of laser is much lower and productivity is much higher.

◆ It is not necessary to build the special support structure for prototype in process because the excess material is in crosshatch shape as is the structure.

◆ It is easy to add the additives to improve the performance of sheet-based materials with mixing the additives in powder state into adhesive before coating the adhesive to sheet.

◆ There is no pollution because the little smoke caused by laser cutting is filtered by the electronic air cleaner.

◆ Appropriate selecting the base-material and matching it with the adhesive and additives will increase the operation temperature of prototype formed by sheet-based materials up to 200℃.

◆ The type of laser used is CO_2 rather than UV. The price of CO_2 laser is much lower and life time is much longer than UV laser.

Obviously the most important direction for material development should be focused on the sheet-based materials.

4. THE EFFECTIVE WAY FOR IMPROVING PERFORMANCE OF SHEET-BASED MATERIALS IS TO MIX ADDITIVES INTO ADHESIVE

The appropriate mixing additives into adhesive will greatly improve the performance of sheet-based materials and prototype. Such as adding the carbon powder into adhesive would cause some advantages as follows:

◆ Higher hardness for surfaces.
◆ Easy to polish, especially for big surface.
◆ Higher heat conductivity.
◆ Easy to separate the excess material from prototype.
◆ Higher wear tolerance.
◆ Higher operation temperature.

The best way to add the additive is to mix it with adhesive and then spray the mixture to the surface of sheet in high-voltage-static field.

Fig.1 is some prototypes made from sheet-based materials with additives and formed by KINERGY's SAMM RPS. From the examples it is can be seen that the complicated and very thin details with smooth surfaces and higher strength in prototypes can be successfully obtained because of the excellent performance of materials.

5. THE FURTHER DIRECTION FOR DEVELOPING SHEET-BASED MATERIALS

To compete with SLA/SLS/FDM and so on, the further direction for developing sheet-based materials will be suggested as follows:

- ◆ Use plastics or compositions of paper and plastics as the base-materials to improve the elasticity and transparency of prototypes.
- ◆ Develop ceramic foil-base-material to form high-operation-temperature-ceramic mold to cast metal parts.
- ◆ Develop metal(aluminum/copper) foil-base-materials to form metal parts or molds/electrode directly.

6. CONCLUSION

The applications and development of sheet-based materials should be emphasized.

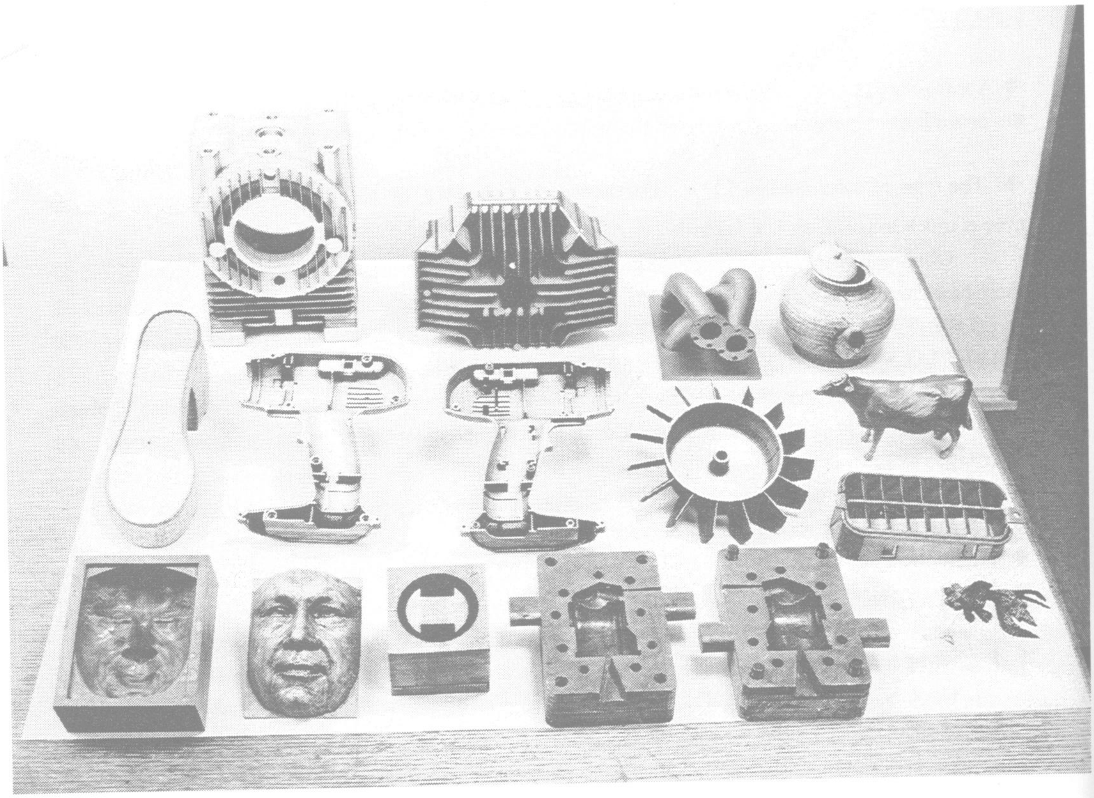


Fig.1 : Some prototypes

FILE FORMAT REQUIREMENTS FOR THE RAPID PROTOTYPING TECHNOLOGIES OF TOMORROW

Jan Helge Bøhn
Department of Mechanical Engineering
Virginia Polytechnic Institute and State University
Blacksburg, VA 24061-0238, USA

ABSTRACT

The .STL file format has become the *de facto* rapid prototyping industry file format. Though not ideal, it is sufficient for today's rapid prototyping technologies, which build mono-material parts. This will not be the case for the rapid prototyping systems of tomorrow. Future systems will enable fabrication of parts with discrete or continuously varying material composition in all three dimensions, and they will enable directional deposition of fiber composites and imbedded sensors and actuators. In either case, the .STL file format will no longer be sufficient, and it will have to be replaced by one that is. The same will be true for CAD systems. Without such changes, we will not be able to realize the full potential of future multi-material automated fabrication systems. This paper examines the reasons for the success of the .STL file format and explores the capabilities that must be present in future CAD systems and file formats to realize forthcoming fabrication capabilities.

KEYWORDS

Rapid prototyping, Layered manufacturing, Solid freeform fabrication, File formats, Future

1. INTRODUCTION

The .STL file format was developed by the Albert Consulting Group for 3D Systems, Inc. to support their new revolutionary fabrication technology; stereolithography [1,2]. This file format did not represent the state-of-the-art in data representation, but, rather, it appears to have been a quick hack put together to make things work. It is hard to believe that it was ever intended for public release or for use in a commercial product. Still, the success of this file format has been impressive. Today, a decade later, the .STL file format remains the *de facto* rapid prototyping industry file format, despite its inefficiencies, inconsistencies, and challenges from competing formats.

The success of the .STL file format is due to its sufficiency, its simplicity, and its initial monopoly. Its mathematical sufficiency stems from it describing rigid solid objects using a boundary representation (B-Rep). Specifically, the .STL file format represents the virtual CAD model of the object to be fabricated as a collection of triangular facets that, when taken together, describe a polyhedral approximation of the object's surface; that is, a polyhedral approximation of the boundary between material and non-material. Since triangular facets can clearly be combined into a triangle mesh that describes such a closed, orientable surface, the .STL file format is therefore mathematically sufficient for describing rigid polyhedral solid objects [3,4,5,6]. However, triangular facets can only provide an approximation of curved surfaces in general, and increasing the accuracy of such approximations come at a cost. The accuracy of a triangle mesh approximation can be increased by reducing the facet size in regions of high curvature [7]. Of course, this increases the number of facets used to describe the object, which in turn increases the associated computer memory, disk space, and processing time requirements. Fortunately, these increased requirements have become less of an issue in recent years, since (a) the sizes of typical .STL files have barely increased, due to the relatively slow growth in fabrication system build envelope size, resolution capability, and speed; while (b) the costs of computer memory, disk space, and processing power have decreased dramatically. Hence, and since .STL files are seldom used for archival purposes but are generated as needed, most users today successfully generate .STL files with triangle mesh resolutions in a range corresponding to the fabrication system resolution. This success demonstrates that the .STL file format is, and continues to be, sufficient for current commercial use.

```
Address  Length  Type     Description
--------------------------------------------------------
  0       80      char     Header information.
 80        4      long     Number of facets in solid.

  First facet (50 bytes):
 84        4       float    Normal (x-component)
 88        4       float    Normal (y-component)
 92        4       float    Normal (z-component)
 96        4       float    Vertex 1 (x-component)
100        4       float    Vertex 1 (y-component)
104        4       float    Vertex 1 (z-component)
108        4       float    Vertex 2 (x-component)
112        4       float    Vertex 2 (y-component)
116        4       float    Vertex 2 (z-component)
120        4       float    Vertex 3 (x-component)
124        4       float    Vertex 3 (y-component)
128        4       float    Vertex 3 (z-component)
132        2       short    Attribute info. (not used)

Second facet (50 bytes):
134 ...
```

Fig. 1 : The .STL binary file format.

```
SOLID Test_Part_567.Manifold_Z
  FACET NORMAL 0.000000e+00 1.000000e+00 2.634149e-09
    OUTER LOOP
      VERTEX 3.000000e+00 1.400000e+00 4.000000e+00
      VERTEX 4.000000e+00 1.400000e+00 4.000000e+00
      VERTEX 4.000000e+00 1.400000e+00 3.000000e+00
    ENDLOOP
  ENDFACET
  FACET NORMAL ...
  :
  < remaining facets >
  :
ENDSOLID
```

Fig. 2 : An example of a file using the .STL ASCII file format.

The success of the .STL file format, however, has also been due to its extreme simplicity. As illustrated in Figures 1 and 2, an .STL file simply consists of a linear list of triangular facets that, when combined, implicitly form the complete triangle mesh. The .STL file format therefore represents a lowest common denominator with respect to other geometric data structures and file formats. Each facet is completely self-contained and has no knowledge of its surroundings. Specifically, there is no direct topological information present, nor are the data elements, such as vertices, indexed. It simply assumes that all the required facets are present and properly described. Hence, as long as this assumption is not violated, the .STL file format is mathematically sufficient in that it can describe any polyhedral, and in that it can implicitly separate regions containing material from those containing no material. Generating an .STL file is therefore a trivial matter, whether it is generated from within a CAD system or converted from another file format. Combining this extremely low cost of implementation with the strategic importance of supporting layered manufacturing, has afforded widespread implementation across CAD systems even though the number of actual users is relatively low. Naturally, the net effect of this evolution has been to reinforce the .STL file format as the *de facto* industry standard.

Lastly, a significant portion of the success of the .STL file format is due to its initial monopoly. During the three years following 3D Systems, Inc.'s 1987 introduction of the stereolithography process, the .STL file format had no rivals and was allowed to develop and mature its position as the *de facto* industry standard. By the time the competing rapid prototyping systems made it to the market, it had become a requirement that they support the .STL file format.

2. CURRENT SHORTCOMINGS OF THE .STL FILE FORMAT

The fact that the .STL file format has succeeded so well in becoming the *de facto* rapid prototyping industry file format does not mean it is without its shortcomings or that alternative file formats have not been proposed. The shortcomings of the .STL file format are well documented [4,5,6,8,9], but until now they have all concerned the efficiency and integrity of the file format and not its sufficiency. This section briefly reviews these current shortcomings.

As discussed in the previous section, the .STL file format has proven it is sufficient for today's near-isotropic, mono-material layered manufacturing technologies. Its only problem with respect to these technologies is that its ASCII version is slightly incompatible with its binary version. One incompatibility is that only the binary version contains attribute information. Fortunately, this difference is immaterial since the use of this information has not been defined (Figure 1). A more serious incompatibility relates to how data is represented in the two .STL file format versions. In the binary version there is no translation of floating point numbers during their transfer from memory to disk and then back to memory again (i.e., when transferring data first from the CAD system to the .STL file, and then to the fabrication system). Hence, this process does not corrupt data. Unfortunately, this is not true for the ASCII version. In the case of the ASCII version, the floating point numbers in memory are first translated into ASCII approximations during their transfer to disk; and then they are read, translated, and approximated again to form floating point numbers in memory. This repeated sequence of approximations is a proven source of data corruption, and it is why, in general, it is best to avoid ASCII file formats to describe floating point numbers.

The remaining shortcomings of the .STL file format, with respect to today's layered manufacturing technologies, are only of inconvenience and do not affect its sufficiency. They include (1) its verbose organization, (2) its lack of topological content to facilitate efficient processing, and (3) its lack of topological information to help ensure that its fundamental assumption is not violated; namely, that all facets are present and correctly positioned. Indeed, missing and erroneous facets have consistently been the main problem experienced with .STL files [10]. The sources of these problems have been, and continue to be, (a) inaccuracies in the data generated by the CAD systems, and (b) invalid designs that do not properly describe rigid solid objects. Much effort has therefore been expended on devising methods for automatic repair of CAD models described in the .STL file format [4,5,6,8,11,12]. Their common approach has been to infer topology during the model reconstruction. While this does not necessarily restore the designer's intent, it does help maintain data integrity by reducing duplication of data; it facilitates efficient processing of data, for instance, during CAD model slicing [13]; and it facilitates more compact data organization, both internally in the processing computer program and in alternative, more efficient file formats [14,15].

3. EFFORTS TO REPLACE THE .STL FILE FORMAT

The inefficiencies and inconsistencies of the .STL file format have given rise to numerous challenges and calls for its replacement. Examples include the use of geometric shapes beyond triangular facets [14,15], the use of triangle facets with topology [14], the use of other standard file formats such as IGES or STEP [14,16, 17], and the generation of two dimensional build layer contours directly from the original CAD model [18,19,20,21]. These proposed replacements address the needs for robustness, algorithmic efficiency, and geometric precision. However, they have failed to provide compelling advantages to justify the marketplace abandoning the .STL file format which has proven its sufficient for today's layered manufacturing technologies.

Still, the current shortcomings of the .STL file format continue to prompt calls for its replacement. Most recently, a National Institute of Standards and Technology (NIST) workshop was held to address these shortcomings, with an eye to define a STEP-based replacement [22]. It was here interesting to note that the .STL file format is no longer sufficient for several of today's research systems, in particular because they no longer limit fabrication to parts of a single near-uniform build material. Multi-material systems require a file format that can describe the locations of the various materials throughout the part being fabricated; and for this, the .STL file format is clearly insufficient. It will therefore be necessary to develop a replacement file format to accommodate the migration of these advances in fabrication capabilities into the market place.

The development of a new replacement file format will have to closely consider forthcoming fabrication capabilities to ensure its commercial success. As a first step in this direction, the following section will examine several forthcoming layered manufacturing technologies and attempt to identify some of the new fabrication capabilities that a future file format should accommodate.

4. FUTURE HARDWARE

Current rapid prototyping systems build mono-material parts, though some systems, including fused deposition modeling (FDM), solid ground curing (SGC), freeform powder molding (FPM), and Sanders Model Maker (SMM), use a secondary material for support structures. For these systems the .STL file format is sufficient since it describes the location of the main material. The support material, if present, is placed in any or all locations not occupied by the main material. Its location is therefore implicitly described in the .STL file format.

Future rapid prototyping systems promise to dramatically increase the range of available automated fabrication capabilities. The following list illustrates some interesting advances that can be expected to emerge and graduate to the commercial market over the next several years:

Internal coloring under computer control — Materialise NV, Leuven, Belgium has demonstrated selective coloring of parts fabricated by stereolithography [23]. A special resin is used that changes its color depending on the local curing conditions.

Fibril direction under computer control — Ogale et al. [24] has demonstrated *in situ* deposition of continuous fibrils into a photosensitive resin. This blending of stereolithography (SLA) and fused deposition modeling (FDM) has the potential of permitting engineers to design parts with locally defined mechanical properties throughout.

Continuously changing material composition under computer control — Stuffle et al. [25] have demonstrated fabrication of parts with continuously changing blend of materials using an FDM-like process. Fitzgerald and Bøhn [26] have explored graded materials for powder-based systems. Similar as with the customized fibrils, graded material compositions can be used to locally manipulate the mechanical, thermal, and electrical properties of a part.

Imbedding foreign components under computer control — Prinz et al. [27] have demonstrated alternating material and component deposition, resulting in parts with pre-fabricated components permanently imbedded. The material deposited could conceivably be any type used for layered manufacturing. The components, on the other hand, are parts that must be, or that are more efficiently, fabricated separately by other means (e.g., electrical components, sensors, and actuators).

Non-planar build layers — Helisys, Inc., Torrance, California, USA are exploring layered object manufacturing (LOM) with non-planar build layers [28]. Curved build layers can be used to improve part strength and stiffness. This is particularly interesting in light of the ongoing development of LOM with sheets of fiber composites [29]. Taking this one step further, Baird, Bøhn, and Stuffle are developing the capability to fabricate parts by extruding continuous fiber composites under full 3-axis motion [work in progress]. This will permit the design of parts with custom fibril direction across non-planar build layers.

This brief sampling of forthcoming advances illustrates how the future will bring exciting new capabilities to the field of layered manufacturing, capabilities which will enable engineers to design parts they earlier only could dream of realizing. Indeed, the real challenge will not be to develop these hardware capabilities, but it will be to the generate computer control that effectively exploits them. The following section will briefly discuss this issue in more detail.

5. FUTURE SOFTWARE

These new forthcoming fabrication processes clearly require too much information to be served by the .STL file format. The same is true about current CAD systems, in that they have been developed to facilitate the design of parts with one or more discrete isotropic or near-isotropic materials. They currently have no means of describing parts incorporating these new forthcoming fabrication capabilities. Nor is it clear what logical and geometric structures would be best suited for describing such parts. It is clear, however, that a future file format will be closely related to these structures the same way .STL, IGES, and STEP are related to B-Reps, since a file format is simply a description of structure. Developing suitable structures is therefore an important area to be researched.

6. CONCLUSIONS

The .STL file format, though not ideal, is sufficient for today's near-isotropic mono-material fabrication systems. However, it is insufficient for several emerging layered manufacturing technologies. A future layered manufacturing file format will have to incorporate details concerning the fabrication process. Examples include multiple discrete materials, continuous material gradients, fiber directions, fabrication across non-planar build layers, surface and internal texture-features, and imbedded components. Floating point numbers will have to be described with double precision to accommodate both increased fabrication detail and build size. Topology must be included for computational efficiency and model integrity. Finally, the file format must be limited to a binary representation to maintain data integrity. An ASCII version containing floating point numbers would be unreliable and would consume needless disk space. ASCII output can always be generated from the binary version for debugging purposes.

7. REFERENCES

1. "Stereolithography Interface Specification," 3D Systems, Inc., Valencia, California, USA, June 1988.
2. A.L. Cohen, ed., "Interfacing CAD with rapid prototyping," *Rapid Prototyping Report*, vol. 2, no. 1, January 1992, pp. 4-6.
3. A.A.G. Requicha, "Representations for Rigid Solids: Theory, Methods, and Systems," *ACM Computing Surveys*, vol. 12, no. 4, December 1980, pp. 437-464.
4. J.H. Bøhn and M.J. Wozny, "Automatic CAD-model Repair: Shell-Closure," *proc., Solid Freeform Fabrication Symposium*, H.L. Marcus et al., eds., The University of Texas at Austin, Texas, USA, 1992, pp. 86-94.
5. J.H. Bøhn and M.J. Wozny, "A Topology-Based Approach for Shell-Closure," in: *Geometric Modeling for Product Realization*, P.R. Wilson et al., eds., Elsevier Science Publishers B.V. (North-Holland), Amsterdam, 1993, pp. 297-319.
6. J.H. Bøhn, *Automatic CAD Model Repair*, Ph.D. thesis, Rensselaer Polytechnic Institute, Troy, New York, USA, August 1993.
7. W.J. Schroeder, J.A. Zarge, and W.E. Lorensen, "Decimation of Triangle Meshes," *Computer Graphics*, vol. 26, no. 2, July 1992, pp. 65-70.
8. J.H. Bøhn, "Removing Zero-Volume Parts from CAD Models for Layered Manufacturing," *IEEE Computer Graphics and Applications*, vol. 15, no. 6, November 1995, pp. 27-34.
9. G.M. Fadel and C. Kirschman, "Accuracy issues in CAD to RP translations," *Rapid Prototyping Journal*, vol. 2, no. 2, 1996, pp. 4-17.
10. L.E. Roscoe, K.L. Chalasani, and T.D. Meyer, "Living with STL files," *proc., International Conference on Rapid Prototyping*, R.P. Chartoff and A.J. Lightman, eds., The University of Dayton, Ohio, USA, 1995, pp. 145-151.
11. I. Mäkelä and A. Dolenc, "Some Efficient Procedures for Correcting Triangulated Models," *proc., Solid Freeform Fabrication Symposium*, H.L. Marcus et al., eds., The University of Texas at Austin, Texas, USA, 1993, pp. 126-134.
12. X. Sheng and I.R. Meier, "Generating Topological Structures for Surface Models," *IEEE Computer Graphics and Applications*, vol. 15, no. 6, November 1995, pp. 35-41.

13. S.J. Rock and M.J. Wozny, "Utilizing Topological Information to Increase Scan Vector Generation Efficiency," *proc., Solid Freeform Fabrication Symposium*, H.L. Marcus et al., eds., The University of Texas at Austin, Texas, USA, 1991, pp. 28-36.

14. S.J. Rock and M.J. Wozny, "A Flexible File Format for Solid Freeform Fabrication," *proc., Solid Freeform Fabrication Symposium*, H.L. Marcus et al., eds., The University of Texas at Austin, Texas, USA, 1991, pp. 1-12.

15. "STH File Format File Description," Brock Rooney & Associates, Inc., Birmingham, Michigan, USA, 1991.

16. P. Carleberg, "Product Model Driven Direct Manufacturing," *proc., Solid Freeform Fabrication Symposium*, H.L. Marcus et al., eds., The University of Texas at Austin, Texas, USA, 1994, pp. 270-276.

17. C.R. Gilman and S.J. Rock, "The Use of STEP to Integrate Design and Solid Freeform Fabrication," *proc., Solid Freeform Fabrication Symposium*, H.L. Marcus et al., eds., The University of Texas at Austin, Texas, USA, 1995, pp. 213-220.

18. S. Guduri, R.H. Crawford, and J.J. Beaman, "A Method to Generate Exact Contour Files for Solid Freeform Fabrication" *proc., Solid Freeform Fabrication Symposium*, H.L. Marcus et al., eds., The University of Texas at Austin, Texas, USA, 1992, pp. 95-101.

19. S. Guduri, R.H. Crawford, and J.J. Beaman, "Direct Generation of Contour Files from Constructive Solid Geometry Representations," *proc., Solid Freeform Fabrication Symposium*, H.L. Marcus et al., eds., The University of Texas at Austin, Texas, USA, 1993, pp. 291-302.

20. R. Jamieson and H. Hacker, "Direct slicing of CAD models for rapid prototyping," *Rapid Prototyping Journal*, vol. 1, no. 2, 1995, pp. 4-12.

21. "Common Layer Interface (CLI)," Brite EuRam Rapid Prototyping Techniques, EOS GmbH, Planegg, Germany, Project BE5278, 1994.

22. H.J. Lee, R.D. Sriram, and M.J. Pratt, "STEP-based Solid Interchange Format Standard," workshop, National Institute of Standards and Technology, Gaithersburg, Maryland, USA, November 25, 1996.

23. B. Kramer, in: JTEC/WTEC Panel Report on Rapid Prototyping in Europe and Japan, vol. II, NTIS report PB96-199583, U.S. Department of Commerce, September 1996, p. 39.

24. C. Greer, J. McLaurin, and A.A. Ogale, "Processing of Carbon Fiber Reinforced Composites by three Dimensional Photolithography," *proc., Solid Freeform Fabrication Symposium*, D.L. Bourell et al., eds., The University of Texas at Austin, Texas, USA, 1996, pp. 307-311.

25. S. Fitzgerald, "A Pneumatic Conveying Powder Delivery System for Continuously Heterogeneous Material Deposition in Solid Freeform Fabrication," M.S. thesis, Virginia Polytechnic Institute and State University, Blacksburg, Virginia, USA, July 1996. URL: http://cadserv.cadlab.vt.edu/bohn/projects/fitzgerald/

26. G.E. Hilmas, J.L. Lombardi, R.A. Hoffman, and K. Stuffle, "Recent Developments in Extrusion Freeform Fabrication (EFF) Utilizing Non-Aqueous Gel Casting Formulations," *proc., Solid Freeform Fabrication Symposium*, D.L. Bourell et al., eds., The University of Texas at Austin, Texas, USA, 1996, pp. 443-450.

27. L. Weiss, F. Prinz, G. Neplotnik, K. Padmanabhan, L. Schultz, and R. Merz, "Shape Deposition Manufacturing of Wearable Computers," *proc., Solid Freeform Fabrication Symposium*, D.L. Bourell et al., eds., The University of Texas at Austin, Texas, USA, 1996, pp. 31-38.

28. G. Kalmanovich, "Curved-Layer Laminated Object Manufacturing," *proc., Solid Freeform Fabrication Symposium*, D.L. Bourell et al., eds., The University of Texas at Austin, Texas, USA, 1996, pp. 273-279.

29. A. Lightman and G. Han, "Laser Cutting of Ceramic Composite Layers," *proc., Solid Freeform Fabrication Symposium*, D.L. Bourell et al., eds., The University of Texas at Austin, Texas, USA, 1996, pp. 291-298.

IMPACT OF RAPID PROTOTYPING ON PRODUCT DEVELOPMENT

R. Noorani, P. Gerencher, M. Mendelson, O. Es-Said, & S. Dormann
Loyola Marymount University, Los Angeles, CA, U.S.A.

B. Fritz
Northrop-Grumman Corporation
Los Angeles, CA, U.S.A.

ABSTRACT

This paper describes the application of Rapid Prototyping technology in the product development and analysis. The Sanders Prototype, a three-dimensional desktop printing rapid prototype machine, has been used to fabricate wax solid models. The form, fit and finish of the products will be compared to those obtained from a 3D Systems machine. The prototype will be cast in bronze using investment casting for further analysis.

The paper will also discuss the theory of rapid prototyping, different types of prototyping machines, methodology of creating prototypes, the limitations and problems associated with the Sanders prototyping machine. A number of design case studies will also be presented. The paper will end with conclusions and recommendation for future trends.

KEYWORDS
Rapid Prototype, Computer-Aided Design, Analysis, Investment Casting, Stereolithography

INTRODUCTION

The keys to regaining competitiveness in most design and manufacturing industries are quality, productivity, reduced costs, customer satisfaction and responsiveness in bringing new products to the marketplace [1-3]. One of the methods for attaining these attributes is Rapid Prototyping (RP). Rapid Prototyping takes information from 3D CAD database and manufactures solid models (prototype) of the design [4-8]. One can turn a design concept into a prototype product, test it for form, fit and function without excessive cost and time of traditional prototyping methods [4-5]. The benefits of RP include (1) reduced lead-times to produce prototype components, (2) improved ability to visualize the part geometry due to its physical existence, (3) earlier detection and reduction of design errors and, (4) increased capability to compute mass properties of components and assemblies [8]. RP is required for the elimination of waste and late design changes before a new product is launched. The integration of RP in product development is shown in Figure 1.

The traditional method of prototyping a part is machining which can require significant lead-times of up to several days to weeks, depending on the complexity of the part and availability of materials, equipment and labor [5]. A number of RP technologies [3-5] such as Stereolithography, Selective Laser Sintering, Fused Deposition Modeling and 3D Printing, etc., are available for producing plastic parts, all of which depend on design data generated on a Computer-Aided Design (CAD) system.

The first commercial prototyping machine was introduced in 1988. Since then the industry had has grown rapidly. Figure 2 shows the growth of rapid prototyping. Figure 3 shows the RP industry as of 1995.

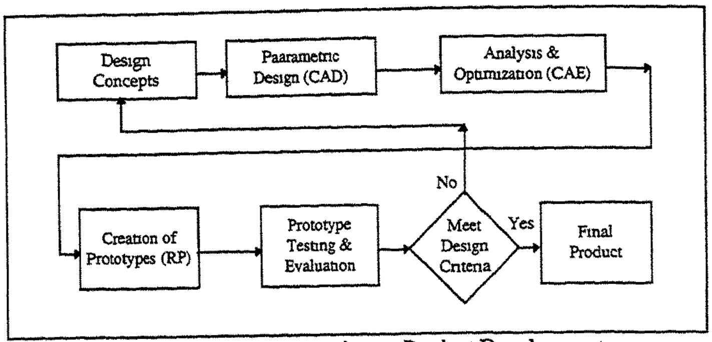


Figure 1: Role of prototyping on Product Development

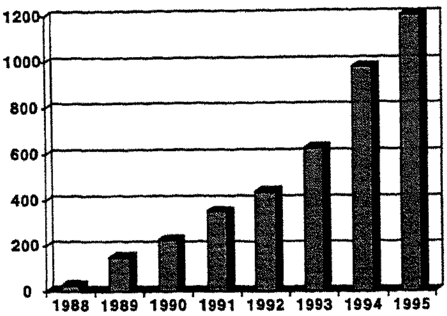


Figure 2. The Growth of Rapid Prototyping

RP System OEM	# of Systems Sold
3D Systems, Inc	645
Stratasys, Inc.	150
Helisys, Inc.	125
DTM Corp.	80
Sanders Prototype, Inc	40
Cubital Ltd.	25
Soligen Inc.	7
Other	183
TOTAL	1255

Figure 3. The RP Industry in 1995

RAPID PROTOTYPING TECHNOLOGY

In this section, a series of new technologies that are available in the market is reviewed. These technologies allow "automatic" production of prototype parts directly from a solid model. The systems discussed here are commercial systems currently available in the United States of America.

3D Systems: The first commercial rapid prototyping machine was introduced in the late 1980's by 3D Systems. 3D Systems introduced the SLA 250 which was the first commercial implementation of the stereolithography system developed by Charles Hull [5]. Stereolithography uses a laser to selectively cure liquid photopolymer creating the desired three dimensional prototype. Soon after 3D Systems entered the market, competing systems began to appear. Some of the new systems were variations on Hull's stereolithography, while others employed completely new technologies.

Cubital: Solid Ground Curing (SGC) is a photopolymer-based system produced by Cubital, an Israeli company. SGC creates a series of photomasks and uses UV lamp to cure an entire layer at once. The technique uses wax as a support structure to hold the part stable during manufacture.

DTM: Selective Laser Sintering (SLS), introduced by DTM, uses a laser to sinter a powdered material into the prototype shape. As the device builds layer after layer, the uncinterd powder acts as support for the part under construction. SLS can produce not only plastic parts, but also has the potential to produce parts from powdered metal or ceramic.

HELISYS: Helisys introduced the Laminated Object Manufacturing (LOM) which creates parts from laminated sheets of paper, plastic or metal. This technique only requires to cut the outline of each layer into a sheet of construction material instead of cross hatching the entire part cross section.

STRATASYS: Fused Deposition Modeling (FDM) was introduced by Stratasys in 1991. FDM extrudes a thin stream of melted polymer or wax through an extruder head whose position is controlled by a computer. Parts are built up by moving the extruder head through the volume of the part.

SANDERS PROTOTYPE, INC. (SPI): SPI introduced 3D Printing technology that uses dual ink printer heads to deposit both thermoplastic part material and a wax support structure. The device uses a milling operation to create very thin, precise layers. Since this research concentrates on SPI's 3D Printing technique, we shall study this process in more details in this paper.

BPM: Ballistic Particle Manufacturing (BPM) was introduced by BPM Technology, Inc. BPM launches small thermoplastic particles at a target where the particles adhere. A computer controls the stream of particles slowly building up a part. While other processes discussed here use three control axes, BPM uses a five-axis head to control the particle trajectories.

SANDERS MODEL MAKER

The Model Maker from *Sanders Prototype, Inc.*, produces parts by printing successive layers on foam substrate attached to an elevator table. Each layer consists of a horizontal cross section of the model. After a layer is completed, the table is lowered slightly, and the next layer is printed on top of the last one. In this manner the thin cross sections are stacked on top of each other to build up the three-dimensional model [9]. Figures 4 and 5 show the 3D printing process and the functional block diagram of the Model Maker.

Many models have an overhang at some point, i.e. a layer that is larger than the one below it. This can create a problem, since there is nothing to support the overhung portion when it is printed; instead, the overhung portion will fall down to the table surface below. To prevent this, temporary support material must be added to the lower layers wherever an overhang will occur. The support material is removed after the model is completed.

Before a model can actually be built, a binary data (BIN) file must be prepared. This file contains the data that describes the physical geometry of the part. It defines each drawn vector of each successive layer of the model and includes both build and support materials.

The Model Maker also requires a build process parameter (BLD) file. This file contains the data that describes how the part will be built. A default file with the most commonly used settings is provided. However, a customized .BLD file can be created to meet particular requirements.

A program called Model Works creates .B1N files from CAD data. This data can be a three-dimensional .STL file, or a series of two dimensional HPGL files (where each file represents a different cross section of the part.) The Model Works program can check the data for errors, allow simple editing, orient the part in various ways, generate support structures, and so forth. Other third party programs can also generate .B1N files for the Model Maker. Some of these programs can also create a customized .BLD file to match each .B1N files.

Once the .BIN file is ready, the Model Maker can build the part. A program called MM reads the .BIN file and controls the machine to print the successive layers. Separate jets are used to print the build and support materials. A jet check station allows the jets to be tested for proper operation, and to be cleaned if necessary, all automatically. As each layer is completed, a cutter is used to trim the surface of the model, to ensure a flat uniform surface for the next layer.

The user can monitor the process through the computer display and a log file that is automatically kept on disk. Modifications to the build process can be made through the computer keyboard and a process parameter file.

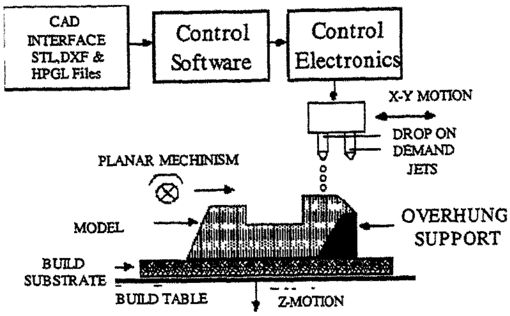


Figure 4. 3D printing process

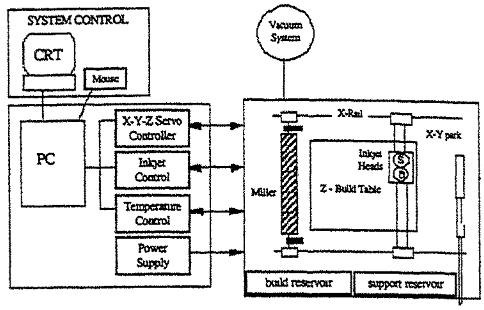


Figure 5. Functional block diagram

The support material is used to support overhangs and cavities in the model during the model build sequence. The build and support materials are deposited onto the build substrate as a series of uniformly spaced "micro-droplets". These droplets may be placed at any desired location upon the build substrate. The droplets adhere to each other during a liquid-to-solid phase transition to form a uniform mass. The drying process is fast enough to allow the milling of the layers immediately following the deposition cycle.

PROCEDURE

The general procedures followed in making the prototypes are as follows;

A. Slicing the part

1. A three-dimensional solid model of the part that is to be produced, is created by either Pro/Engineer, AutoCAD or any other comparable software package.
2. The model must then be saved in any of the following CAD formats:
   a. .STL, (ASCII or binary)
   b. AutoCAD .DXF
   c. 3D System's SLC
   d. 3D System's HPGL

   These files can be read and loaded into the "Model-Works" software provided with the RP machine.
3. Once the model is loaded, the model is ready to be sliced into layers for the prototyping process. The slice configuration is defined by the user. Variables in the configuration include: slice thickness, printing pattern density, extra support material, extra cooling time, and jet check parameters.
4. Once loaded, the part can be viewed as it will be built. At this point the part is moved to the origin of the building surface. The part can also be rotated in any angle until the user is satisfied with the building orientation.
5. The model is then sliced using the defined configuration and is ready for the prototyping process.

B. Building the model

1. A piece of foam substrate is placed onto the build table. The substrate is then milled until a sufficiently smooth surface is attained.
2. The machine is then allowed to proceed with the model making process. This process is

fully automated and will continue until the model is built.

3. The finished model contains support material that must be removed. This is done by placing the model in solution to dissolve the support material and attached substrate.

RESULT

Of the several prototypes that were made using the Sanders Prototype (Model Maker), only two will be presented here. Figure 6 shows the first part. The part was created in a Sun Microsystems's Sparcstation 5 computer using Pro/Engineer solid modeling software. The .STL file was then transferred to Model Maker where it was prototyped. Figure 7 shows a complex mechanical part. Because the part came from a defense farm, we cannot release the name of the part. The form, fit and function of both the parts appear excellent. Using Coordinate Measuring Machine (CMM), the dimensional accuracy of parts produced using the model maker were measured and analyzed. The measurement and analysis were done at Jet Propulsion Laboratory (JPL) of NASA. We have not received the results of the analysis yet. We expect to present the final result during the conference presentation in April, 1997.

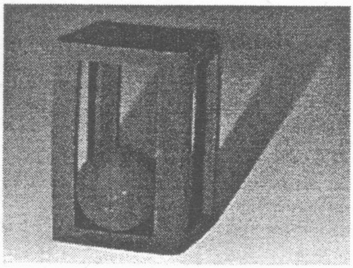


Figure 6: Part-1

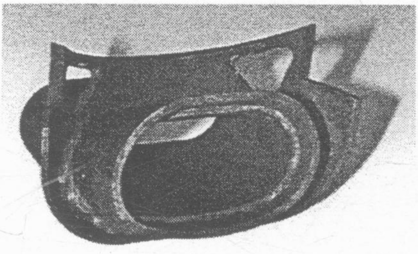


Figure 7: Part-2

The prototypes made by the Sanders Prototype were compared to those made by the 3D System's stereolithography process. Except for the material strength, the prototypes from the Sanders prototype compared favorably with those from 3D Systems in so far as the form, fit and finish were concerned.

CONCLUSIONS

This paper has shown that RP is a new technology that is growing very rapidly and helping many industries justify the cost of the implementation of CAD/CAM/CAE. The major application of RP is in the early verification of product design and fabrication of prototypes for testing. The solid model that RP produces can be used for investment casting.

Although the RP machine worked very well most of the time, there are times when we encountered problems that had to be fixed to keep the machine running. Some of the problems we encountered are jet failures, clogging of filters, failure to seal build tank, software crash and the flat plate syndrome in which the flat parts tend to chip away from the substrate early in the prototyping process.

Overall, the rapid prototyping experience has been very exciting for both the students and the teachers. More and more industries are buying RP machines in record numbers. The RP industry already represents nearly a half-billion dollars in products and services.

RP is being driven by the need to design new products faster than ever before. When the new tools of solid modeling and RP are integrated properly into a concurrent engineering environment, it can make a difference between success and failure. The future of this rapid prototyping industry is very promising.

REFERENCES

1. U. Menon. "Strategies to Promote Concurrent Engineering in Design Classes". Proceeding of the ASEE Conference-PSW Section, Loyola Marymount University, Los Angeles, CA, October 14-16, 1992, pp. 181-192.
2. Ashley, S. "Rapid Prototyping Systems- Special Report", Mechanical Engineering, 1991, pp. 34-43.
3. Ashley, S. "Rapid Prototyping is Coming of Age". Mechanical Engineering, July, 1995, pp.62-68.
4. Jacobs, P. Rapid Prototyping and Manufacturing & Fundamentals of Stereolithography. SME Publications, 1992.
5. Thomas, C.L. An Introduction to Rapid prototyping.. SDC Publications, Mission, Kansas, USA, 1996.
6. Ledbetter, D.B. "Rapid Prototyping 101". Proceedings of the 9th Santa Fe Symposium on Jewelry Manufacturing Technology, Albuquerque, NM, May, 21-24, 1995, pp. 11-33.
7. Wohlers, T. "AutoCAD and Rapid Prototyping". CADENCE, November, 1995. pp. 45-60.
8. Groover, M. Fundamentals of Modern Manufacturing, Prentice-Hall, Inc., 1996.
9. Model Maker Reference Manual, Sanders Prototype, Inc., Pine Valley Mill, Wilton, NH.

DIRECT SLICING FOR RAPID PROTOTYPING

Huang Qiming and Wang Yungan
National Key of The Molding & Technology
Huazhong Univ. of Sci. & Tech. P.R.China

ABSTRACT

Now, most CAD systems could transfer the 3D models into tessellated STL files, which are input datas for rapid prototyping (RP) system. In this way, slicing algorithm is unified but at same time it also results in two major disadvantages: low accuracy and tendency to error. However, most CAD systems have slicing instruction, we could slice the 3D model via this instruction and get the slice data which drive RP system. This method is called direct slicing. It would overcome the inherent disadvantages of STL files.

KEYWORDS: direct slicing, Tesselation, RP system, CAD

1. INTRODUCTION

With the engineerimg application of RP system, how to raise the product accuracy is an important problem to solve. The method of mapping a triangular mesh over a surface or solid model (tessellation) was devised. It offers the possibility of an easy input from CAD system to RP system. However, this method results in many disadvantages. Direct slicing the 3D models in CAD systems would raise the product accuracy.

2. TESSELLATION AND DIRECT SLICING

2.1 Tessellation

The principle of tessellation involves approximating 3D shapes with a "carpet" of triangular patches. Many CAD systems have implemented STL format file, which has become the de facto standard but has disadvantages include the following:

- The STL file carries a high degree of redundancy since each triangle is individually recorded and shared ordinates are duplicates within a file.
- The implementation of STL translators within CAD systems varies and consistency of quality is a problem. This has given rise to "repair software" which slows the production cycle time.
- The subsequent slicing of large STL files can take many hours and, except for RP processes which can slice while they are building the previous layer.
- Occasionally, the designer will be unable to get the CAD model through the STL interface successfully, resulting in remodelling.

2.2 Direct Slicing

An alternative to using an intermediary tessellation file is to slice the CAD model directly and transfer the resultant contours to the RP process.

If RP models were only being used for design visualization, designers would be content with an approximate physical representation of their CAD models. However, with RP models being used increasingly for engineering applications, coupled with the overall requirement for model integrity, it is considered of strategic importance to many designers that the integrity of this model is maintained throughout the process from the concept to the final product.

There are a number of reasons for using direct slicing, mainly related to the disadvantages of using tessellation:

- reduced file size;
- greater model accuracy;
- reduced RP machine pre-processing time;
- elimination of repair routines .

Howevert, there are also potential disadvanteges of direct slicing which include the following:

- beam compensation and offsets still require processing.
- More designer knowledge is required.

3. DIRECT SLICING OF CAD DATA

Auto CAD software can be accessed directly by a language such as C & AutoLISP to access its underlying programs. In this way, it is easy to slice a solid using software calls. These slices can drive the RP systems via drives.

3.1 Slicing 3D Model

Slicing is to get the intersection line of a fascet and an object.It is the basic operating of CAD.

By simply implementing SOLMASSP, we can get the range of the solid model. After getting the slicing position, we can implement SOLSECT, the profile instruction provided by Auto CAD, to slice the 3D model to get the data of the slice.

3.2 Drawing Grids

For the convenience of getting rid of the wastes, it is necessary to draw grids on the useless part of the slice. To do this, one has to solve two problems beforehand:

1.How to get the cross points of the grids & the contour of the slice.

2. How to descrimate between outside & inside contour of the slice.

For problem1, Auto CAD provides instruction EXPLOD in which polylines, made of line mini_segments, are enployed to approximate plane curves. Thus, we can easily solve problem 1 by getting the cross points of grid and these line mini_segments.

As for problem 2, one can use BREAK instruction in Auto CAD to intersect each line of the grid at its crosspoint with the outour line to divide it into several segments. Then by numbering the segment and only keeping those with odd number, one will get the aquired grid.

3.3 Driving RP

There are two ways to drive RP. One is to convert the slices to CLI or HPGL format file, the other is to write their own machine driver. We choose the later. We convert the slices and grids to a DXF file of which the data will drive RP via the driver.

4. TESTING

To ensure the usability of the program, a number of tests were carried out. The slicing procedure was tested using several different parts. The parts were sliced and displayed in a Auto CAD viewer to ensure they were correct. Features that were tested in the parts inclue:

* parts with multiple section results at a given height (Figure 1), this creates the need for several sheets to be created for one layer.

* horizontal holes in parts(Figure 2), making sure the slicing takes place exactly on the bottom or the top of the hole without program failure;

5. CONCLUISIONS

The work started has proved successful and will continue. It has been shown that direct slicing can be beneficial in terms of file size and in cutting out the need to slice a tessellated equivalent model. Accuracy can be enhanced, esapecially on rounded or tubular designs, which also benefit from reduced processing time before the build can start. The ability to perform direct slicing on a particular CAD system has an appeal to certain industry sectors.

It is quite likely that all the major CAD system vendors will develop direct and adaptive slicers in the future since they are a useful tool for supporting RP and no one will wish to be seen behind their competitors in this growing field.

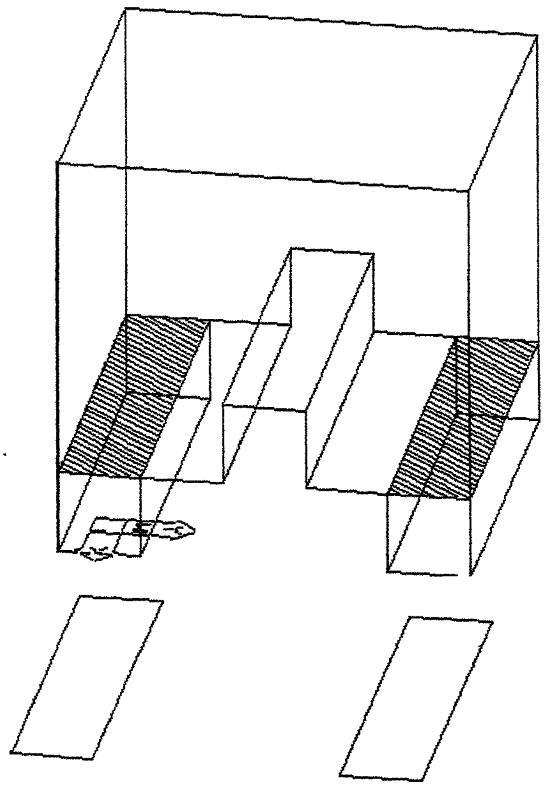


Figure 1. Part with two results obtained from one section

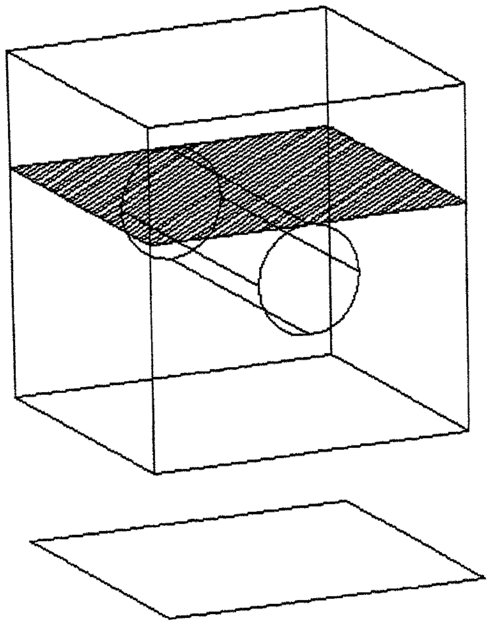


Figure 2. Creation of a non-mainfold body by sectioning

6.REFERENCES

1.Ron Jamieson and Herbert Hacker, Rapid Prototyping Journal, pp.4-12, MCB University Press, 1995.

PARAMETER TUNING AND OPTIMIZATION FOR SLA RAPID PROTOTYPING MANUFACTURING PROCESSES

Jack G. Zhou and Daniel Herscovici
Department of Mechanical Engineering and Mechanics
Drexel University, Philadelphia, PA, USA
and
Calvin C. Chen
Bell Labs, Lucent Technologies Inc.
Murray Hill, NJ, USA

ABSTRACT

The functional requirements of a rapid prototyping system are speed and accuracy, and they both are functions of vendor defaulted and user selected manufacturing parameters. Accuracy is examined by dimensional errors, form errors and surface roughness of manufactured parts. A special designed specimen with twenty dimensional, geometry, and surface roughness features has been used in the inspection of RP manufacturing processes. In terms of Taguchi experimental design techniques, an orthogonal array of experiments has been developed which has the least number of experimental runs and desired process parameter settings. Using the 3-D coordinate measuring machine and the surface topography measurement system, a series of measurements in evaluating the SLA parts quality has been conducted to find the functional relationships between the output part quality and input manufacturing process parameters. Two analysis tools, response surface methodology and ANOVA, have been used to evaluate the SLA RP process and to perform the product optimization.

KEYWORDS

Rapid Prototyping, Stereolithography, Parameter Tuning, Taguchi Method, Response Surface Method

1. INTRODUCTION

In 1987, 3D Systems introduced the first commercialized Rapid Prototyping (RP) system based on Stereolithography (SL). Since then, RP has grown into an integral part of new product development. The use of RP has reduced time to market, cut total costs and improved product quality by giving design and manufacturing teams the opportunity to verify and fine tune designs before committing them to expensive tooling and fabrication. Today SL remains the most popular RP technique with the largest installed user base in the world. RP is a material additive process, where a 3D computer model is sliced and reassembled in real space, layer by layer. In the SL process, this is accomplished using a UV laser to trace and cure successive layers on a photosensitive resin. The virtual part is rebuilt in real space in a fraction of the time needed to create a "traditional" prototype. Although RP has evolved so fast and received more and more attention in these years, it is still in its developing stage, and some critical research problems affect RP's survival, maturation and further applications [1]. In particular, there exists one problem that deserves high priority and needs to be addressed urgently, which is 'how to improve RP part accuracy/quality to be comparable to the conventional machined parts?'

2. ACCURACY PROBLEMS IN THE SLA RP PROCESSES AND ITS IMPROVEMENT

RP processes are integrated manufacturing processes which include CAD, CAM, control of laser devices, materials, manufacturing parameter setup, and postprocessing. Individual process can introduce errors one way or another [2, 3, 4, 5], as explained below. These errors severely reduce RP product accuracy and obstruct its further applications in rapid tooling and functional part fabrication.

1. *CAD/CAM induced error.* Most rapid prototyping systems use the *de facto* standard STL CAD file format of solid representation to define parts to be built. However, STL files pose the problems of dimension, form and surface errors resulting from approximation of three dimensional surfaces by triangular facets. Although a large number of facets can be used to reduce these errors, doing so will result in a giant data file, and longer part build time. 2. *Laser beam width induced error.* The laser beam used to create parts is of a finite width, though the file used to drive the machine represents the edges as zero-width lines. The width of this beam can be compensated for in the laser beam scan control software, but the beam width is not constant from machine to machine and even not same on a single machine over time. This induces part errors. 3. *Material shrinkage error.* SLA parts accuracy is a direct result of the resin properties. Many researchers are striving to develop new resins that offer low shrinkage and high dimensional stability. The earlier resins available from 3D Systems Inc. are primarily limited to the Acrylate base resins with relatively large shrinkage (5%-7% in volume), causing severe distortions of the finished parts. 4. *RP Machine Parameter Setup.* Errors that occur during the building time are mainly in the manufacturing control factor setup which are RP machine vendor defaulted and user selected parameters. Different parameter setup will generate different machining accuracy and build time. 5. *Postprocessing error.* SL parts are designed to be postcured as soon as they are built, otherwise green creep distortion, which results from the residual internal stress generated during the SLA building cycle, will occur. An accumulation of the above five errors usually causes 250-500 μm dimensional error and very unpleasant surface roughness, which make RP products unacceptable in many applications.

Fortunately in the last four years, many research efforts have been dedicated to RP technology and the accuracy problems have been improved significantly. In CAD/CAM a new SLC file format was defined by 3D Systems in 1992. Unlike the tessellated solid STL representation, it is a 2-1/2D contour representation of the model boundaries within each layer. SLC data can be generated from various sources such as CAD solid or surface models or CT scanners. The SLC file format can virtually eliminate the problem of translating a part from the original representation to an intermediate tessellation form, where the error is induced. Some other approaches to eliminating STL format error are also under investigation such as NURBS and direct slicing methods [6]. In laser beam width, a better laser beam control mechanism and beam width compensation software has been used, which reduces the error to a minimum. On the materials side, in early 1994 a significant advance was accomplished when 3D Systems and Ciba Geigy Corporation introduced an Epoxy resin called XB5170. Unlike the Acrylate resin, the small shrinkage of XB5170 (2%-3% in volume) makes it possible to build a stable and accurate part. In postprocessing, along with XB5170 resin, 3D Systems also developed a new part build style ACES (Accurate, Clear, Epoxy, Solid). This is accomplished by completing and uniforming polymerization during the part building process, virtually eliminating postcure distortion and internal stresses. The only problem left on which not much progress has been made is RP machine parameter setup.

3. OBJECTIVES OF THIS STUDY

This research aims at conducting a scientific and experimental study on improving RP parts accuracy through parameter tuning and optimization of SLA manufacturing processes. The optimization results or technology obtained from SLA can be applied to any other RP machines using similar approaches. The objectives of the research are (1) to conduct a detailed experimental study and to search for the inner relationship between SLA RP products accuracy and the machine parameter setup by using Taguchi experimental design, and parameter tuning and optimization techniques, (2) to achieve an accuracy of less than 50 μm error in the X, Y and Z dimensions, and form features, respectively, and a surface roughness around 0.5 μm for top surface and 6 μm for layered surfaces.

4. RESEARCH APPROACHES

4.1. Enhancing Accuracy Using Parameter Tuning and Optimization for SLA RP Processes

In the Stereolithography Apparatus (SLA) there are about fifteen variables to consider prior to a manufacturing process (e.g., layer thickness, Z level wait, sweep period, over cure, blade gap, hatch spacing, dip velocity and dip acceleration). Each of these variables affects the output part quality and build time in some way [2]. Through initial observations and experience with the SLA system it has been found that although most controllable factors have some effects, a major part of output quality is dictated by a few primary control factors [7], such as layer thickness, hatch spacing, overcure, blade gap, and position on the build plane. In the present investigation, we have included these five build parameters as the main factors, and examine their effects on dimensional accuracy, form error and surface roughness of the final parts, individually or interactively. A standard sample has been designed which contains twenty separate features for inspection. Using this sample, more than 100 SLA parts, including 27 different runs, and trial and error statistical analysis, were built. The experimental data were gathered using a Brown and Sharpe Coordinate Measuring Machine (CMM). Based on Taguchi's experimental design, two analysis tools, response surface methodology and ANOVA, have been used to evaluate the SLA RP processes and to perform product optimization. In the following, the new approaches will be discussed in detail.

4.2 Standard Part Design

In order to properly define accuracy, a standard sample needs to be designed to represent some common dimensional/geometric features and surface roughness for SL part accuracy evaluation. The designed part can be depicted in Figure 1, which includes twenty dimensional, geometric and surface features, and can be classified as four groups: the horizontal (X/Y) dimensions, the vertical (Z) dimensions, the geometric forms, and the surface roughness (see Table 1). It should be noted that the vertical (Z) direction is the layer build direction, where a larger amount of dimensional inaccuracy is often observed in comparison with the horizontal (X/Y) dimensions [4]. Note also that the dimensions assigned to the individual features are all the exact multiples of 0.036 inches, a common multiple of the three different layer thickness used in the experiments (i.e., 0.006, 0.009, and 0.012 inches), so that any round-off errors resulting from layer slicing can be completely eliminated.

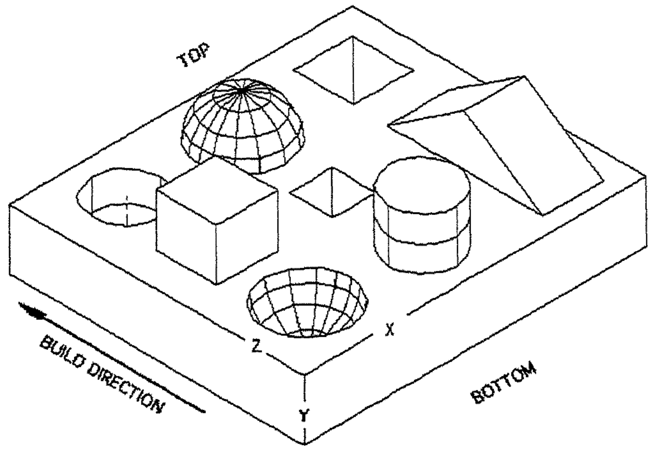


Fig 1. A standard sample with multi-features of dimensions, forms, and surface roughness

4.3 Measurement Technique

In order to reduce human error and to guarantee the resolution of the measurements, a Brown and Sharpe Coordinate Measuring Machine (CMM) was used. The Coordinate Measuring Machine is a reliable and high performance inspection station with programmable flexible gages for process control and with high speed Direct Computer Control capabilities. The Controller servo board is capable of 0.001 mm resolution at speeds up to 175 mm/sec per axis. The machine is certified with an accuracy of 6.35 μm over 255 mm and a repeatability of 3.1 μm. Another advantage of the CMM is that it allows unattended batch measurements, semi-automating the measurement process. In order to assure the integrity of each part measurement procedure, a new local alignment system was constructed for every part. Therefore error in placing the parts on the batch sheet did not contribute any errors to the measurements. Figure 2 is a photograph showing the batch measurement and local alignment techniques.

Table 1. The twenty dimensional, geometric and surface features on the standard part

Group Name	Symbol	Description	Group Name	Symbol	Description
Horizontal X/Y	X1	Large square hole X length	Vertical Z	Z1	Large square hole Z height
Dimensions	X2	Round hole X diameter	Dimensions	Z2	Square protrusion Z height
	Y1	Square protrusion Y width		Z3	Round hole Z diameter
	Y2	Round protrusion Y width		Z4	Round protrusion Z diameter
Geometric	R1	Roundness of the hole	Surface	TOP	Top surface of the base
Forms	R2	Roundness of the protrusion	Roughness	SPT	Square protrusion top surface
	ANG1	Top slope of the triangle		RPS	Round protrusion end surface
	ANG2	Bottom slope of the triangle		SSB	Side surface of the base
	ES	Sphereness of external sphere		FSB	Front surface of the base
	IS	Sphereness of internal sphere			
	FLA	Flatness of the base surface			

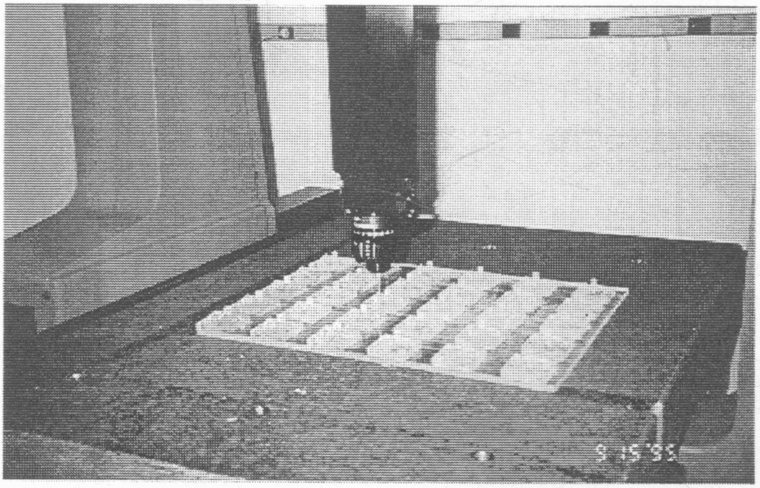


Figure 2. Batch measurement and local alignment techniques used in the CMM measurement

4.4 Experimental Design Using Taguchi Method

A standard approach for experimental design is to use the full factorial method. However, a full factorial approach is acceptable only when a few (usually not more than three) factors are to be investigated. for example, a full factorial approach requires a total of 3^f experimental runs if there are f number of factors to be investigated, each consists of 3 different levels (i.e., high, medium, and low). In the present study, we are searching for the effects of five controllable factors, as well as the interactions between them, if there is any. Application of the full factorial approach to the present investigation requires a total of 243 experimental runs, and is, therefore, not practical. Instead of the full factorial approach, Taguchi [8] has developed a family of Fractional Factorial Experiments to minimize the number of total experimental runs. These Fractional Factorials are orthogonal in nature, based on a set of orthogonal linear equations, and, therefore, allows experimenters to isolate individual factors and use fewer experimental runs [9, 10]. Using Taguchi technique, we are able to reduce the 243 experimental runs to only 27 runs to study the effects of these five control factors including the effects of interactions of any three variables, each at three different levels. Our intentions are not only to improve individual part quality but also to minimize the variance from part to part. The experimental set-up is shown in Table 2, according to Taguchi's L27 Orthogonal Array.

In order to obtain a meaningful measurement result, four identical parts were built for each case, using a 3D Systems' SLA-250 machine, which results in a total of 108 parts. Approximately 100 points on each part were measured using the CMM machine for every dimension under consideration, and the measurement results were carefully screened statistically to eliminate any extraordinary points caused by contamination or other reasons during the building processes. Note that in Table 2, all units used are in mills except for the position where I, M, and O represent the parts location at inner, middle and outer portion of the platform, each occupies approximately 1/3 of the total platform area respectively. The scan time and recoat time in Table 2 are not control factors, but they are listed for latter build time consideration. Note also that, in order to obtain the same amount of resultant overcure as specified in Table 2, the input values of hatch overcure to the SLA part's preparation program need to be different pending on the hatch space and number of scan paths used in the processes. These values can be calculated according to the average exposure versus cure depth equation [4,11] and the corresponding values (mostly negative) are shown in brackets in Table 2

Table 2. The Designed Orthogonal Array for the Experiment Setups (Unit: mil = 0.0254 mm)

Run #	Layer Thickness mil	Resultant Overcure mil	Hatch Spacing mil	Blade Gap mil	Position	Scan Time hr.	Recoat Time hr.
1	6	1 (-4 6, one hatch)	2	1	I	0.400	3.622
2	6	1 (-4.6, two hatches)	4	5	M	0.402	3.622
3	6	1 (-0.5, two hatches)	10	10	O	0.369	3.622
4	6	2.5 (-3.1, one hatch)	2	10	O	0.521	3.622
5	6	2.5 (-3.1, two hatches)	4	1	I	0.518	3.622
6	6	2.5 (1.0, two hatches)	10	5	M	0.507	3.622
7	6	4 (-4.8, two hatches)	2	5	M	0.695	3.622
8	6	4 (-1.6, two hatches)	4	10	O	0.687	3.622
9	6	4 (2.5, two hatches)	10	1	I	0.671	3.622
10	9	1 (-7.8, two hatches)	2	5	O	0.536	2.415
11	9	1 (-4.6, two hatches)	4	10	I	0.521	2.415
12	9	1 (-0.5, two hatches)	10	1	M	0.515	2 415
13	9	2.5 (-6.3, two hatches)	2	1	M	0.688	2.415
14	9	2.5 (-3.1, two hatches)	4	5	O	0.690	2.415
15	9	2.5 (1.0, two hatches)	10	10	I	0.656	2.415
16	9	4 (-4.8, two hatches)	2	10	I	0.902	2.415
17	9	4 (-1.6, two hatches)	4	1	M	0.883	2.415
18	9	4 (2.5, two hatches)	10	5	O	0.849	2.415
19	12	1 (-7.8, two hatches)	2	10	M	0.771	1.811
20	12	1 (-4.6, two hatches)	4	1	O	0.761	1.811
21	12	1 (-0.5, two hatches)	10	5	I	0.727	1.811
22	12	2.5 (-6.3, two hatches)	2	5	I	0.961	1.811
23	12	2.5 (-3.1, two hatches)	4	10	M	0.903	1.811
24	12	2.5 (1.0, two hatches)	10	1	O	0.940	1.811
25	12	4 (-4.8, two hatches)	2	1	O	1.258	1.811
26	12	4 (-1.6, two hatches)	4	5	I	1.252	1.811
27	12	4 (2.5, two hatches)	10	10	M	1.217	1.811

4.5 Statistic Analysis and Optimization Using the Response Surface Method and ANOVA

In many technical fields, there is a relationship between a response variable y of interest and a set of controllable variables $\{x_1, x_2,...,x_n\}$. In some systems, the nature of the relationship between y and the x's might be known exactly, based on the underlying scientific principles. Then one can write a model of the form $y = g(x_1, x_2,..., x_n) + \varepsilon$, where the term ε represents the error in the system. This type of relationship is called a mechanistic model. However, in most practical situations, the underlying mechanism is either unknown or difficult to describe completely, and the experimenter (scientist or engineer) must approximate the unknown function $g(.)$ with an appropriate empirical model $\hat{y} = f(x_1, x_2, ..., x_n) + \varepsilon$. Usually the function $f(.)$ is a first-order or second-order polynomial. This empirical model is called a response surface model [12].

Multiple variable regression is used to develop the empirical model by determining an optimal set of model coefficients. After finding the optimal set of model coefficients, the empirical model $f(.)$ is obtained. Then the analysis of variance (ANOVA) method is used to evaluate the confidence interval and adequacy of the model $f(.)$. We can use ANOVA method to test the significance of the empirical model.

The Response Surface Methodology can be used to find the levels of the controllable variables that result in optimization of the response variable, or discover what levels for the x's will result in a product (process) satisfying several requirements or specifications.

5. EXPERIMENTAL RESULTS, ANALYSIS AND DISCUSSION

As stated above, a total of 20 dimensional, geometry and surface features were defined for the standard part. The test parts were manufactured using a SLA-250 rapid prototyping machine, according to the individual build parameters shown in Table 2. After manufacturing of the total of 108 parts (4 parts each for 27 different experimental runs), detailed measurements were then taken using the Brown and Shape Coordinate Measurement Machine. Each feature consists of a total of approximately 400 measurements taken from each of the four parts built under the same conditions. Some of these measurement values were discarded if they were outside the valid range of the corresponding statistical distribution curve, possibly due to the contamination during the manufacturing and cleaning processes. For the cases of low overcure, layer de-laminations were sometimes observed along the bottom face of square holes or protrusions due to the extreme low overcure used, and these data were also discarded. The remaining valid measurements were then analyzed using the Response Surface and ANOVA methods to conduct the parameter tuning and optimization process. The results obtained can be best illustrated using the following two examples, as shown in Table 3 and Figure 3.

Table 3. Tests on the five factors and three interactions for the large square hole Z height

Factor	Coefficient	Standard dev	t-ratio	Probability α	R^2	$(R^2)_{adj}$
Layer Thickness	0.000026	0.000105	0.248	0.805		
Overcure	-0.001354	0.000105	-12.836	<0.0005		
Hatch Spacing	0.000335	0.000102	3.293	0.001		
Blade Gap	-0.000266	0.000150	-1.772	0.080	91.9%	90 1%
Position	-0.000964	0.000161	-5.973	<0.0005		
LT x OC	0.001179	0.000157	7.515	<0.0005		
LT x HP	0.001978	0.000151	-13.115	<0.0005		
OC x HP	0.001704	0.000169	10.091	<0.0005		

Table 3 illustrates the analysis results to be utilized for finding the significance of the five control factors and three interactions affecting the accuracy of the vertical Z height of the large square hole. In Table 3, the t-ratio represents the statistic t-test value for individual regression coefficient. The larger the absolute value of t-ratio the more significant the factor plays. The probability α value represents the factor coefficient, and the smaller the value the more significant it represents (In the present analysis, the threshold value was chosen at 0.05). If the α value is greater than the chosen level, the null hypothesis is then accepted and the coefficient is judged not significant. Finally, the values of R^2 and $(R^2)_{adj}$ represent the regression confidence and the adjusted regression confidence respectively. They should be at least as large as 60%, and the two values should not differ too much [12]. In view of the analysis results shown in Table 3, it can be concluded that the confidence of the regression model is satisfactory with a value larger than 90%, and the overcure has the most effects on the vertical Z dimension, which has the largest absolute value of t-ratio (not including interactions), with a probability α value of less than 0.0005. In addition to the analysis results, the effects of the individual factors can also be plotted , using the MINITAB software. Figure 3 gives examples of the MINITAB plots, in which the effects of the various factors on the vertical Z height of the large square hole at different levels are revealed, and the optimal parameter setup can be easily realized. Note that in Figure 3, the three points shown in each plot of the five control factors denote the corresponding error values when such factor is set at low, medium and high level respectively. Realizing that our goal is to reduce the dimensional error to the smallest value possible, it can be easily concluded that, from Figure 3, the best setup of the build parameters for making prototypes for large square holes (in fact, any rectangular holes) is to use a medium layer thickness (0.009 inches), a medium hatch space (0.004 inches), a medium blade gap (0.005 inches), a low value of resultant overcure without causing the undesired layer de-lamination (0.001 inches, if possible), and by placing the part at the inner portion of the platform (e.g., at the center of the platform). Finally, it is worthy mentioning that the above recommended setup is only valid for a particular build in which all the features involved are the various sizes of square or rectangular holes. Any other geometrical dimensions and/or different geometrical forms must be excluded from this build, since they may require a different setup of the various factors for achieving the optimal results. An optimal setup for the cases of mixed geometrical forms and dimensions can only be suggested after all the twenty features are examined and results carefully analyzed. Following the procedures mentioned above, all the twenty dimensional, geometry and surface features specified in Table 1 of the standard sample were examined. The measurement results from each of the 108 parts were carefully screened, and the remaining valid measurements were then analyzed according to RSM and ANOVA methods, together with the Main Effect Plots obtained from MINITAB. From these analyses, we were able to reach the conclusion of the main factors and best setup of the build parameters for each of the twenty dimension and form features. Such findings are summarized in Table 4, in which we also list the suggested setting level of the main factors for achieving the best results.

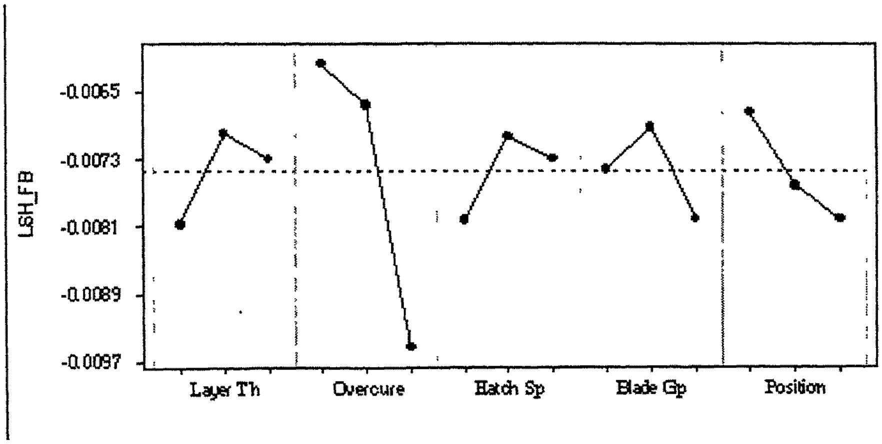


Figure 3. The main effects plot for the large square hole Z height

The analyses results shown in Table 4 gives the SLA users a guideline for building accurate stereolithography parts when a specific dimensional, geometry or surface feature is encountered. It is recognized, however, that most stereolithography parts involve many features listed above, and since every one of them is equally important, a suggested optimal setting of the various build parameter for the general parts is deemed very useful. We, therefore, carefully examined our experimental and analysis results, and concluded that, for a general part, the best setting to achieve an over-all high dimensional, geometry and surface accuracy is to use low layer thickness (0.006 inches or less), low resultant overcure (0.001 inch), medium blade gap (0.005 inches), medium or low hatch space (0.004 or 0.002 inches), and medium or low position on the platform (i.e., in the center or close to the center of the platform).

Table 4. Summary of main factors and the corresponding best setting for the various dimensional, geometry and surface features

Group Name	Symbol	Major Factors & Best Setting	Group Name	Symbol	Major Factors & Best Setting
Horizontal X/Y Dimensions	X1	For all the X/Y dimensions,	Vertical Z Dimensions	Z1	Overcure has the largest effect.
	X2	the blade-gap has the largest		Z2	Low overcure is the best.
	Y1	effect on accuracy. Medium		Z3	Layer thickness has the largest
	Y2	blade-gap is the best.		Z4	effect. Low setting is the best.
Geometric Forms	R1	Layer thickness has the	Surface Roughness	TOP	Overcure and layer thickness
	R2	largest effect. Low is the best.		SPT	both are significant. Low best.
	ANG1	No major factors. Confidence is less than 50%		RPS	For all the three features layer thickness has the largest effect
	ANG2	For all the four features		SSB	Low thickness is the best.
	ES	Layer thickness is significant.		FSB	
	IS	Low thickness is the best.			
	FLA				

6. BUILD TIME CONSIDERATIONS

Since the total build time required for a specific part depends on the total recoating time consumed between layers and the laser scan time used at each layer, and since both of them are strong functions of part's size and geometry, the optimal build time obtained from the present experiments serves little use and may lead to a wrongful conclusion when applied to other parts of different sizes and geometry. Instead, we should compare the total laser scan time and the total recoating time separately. The saving due to laser scan time can be significant when building large cross section area parts, and the saving in total recoating time is important for building long and tall parts. For the present cases, the total laser scan time required to build one part for each of the 27 runs can be calculated using the build time prediction program written by Chen and Sullivan (1996), and the corresponding calculated results, based on an assumed laser power of 25 mW, are shown in Table 2. Table 2 also gives the calculated total recoating time for each case, which reveals a saving of factors 1.5 and 2 respectively when using 0.009 inches and 0.012 inches layer thickness instead of the 0.006 inches layer thickness normally used in ACES build. As expected, no significant difference in total laser scan time was found when applying different hatch space to the parts building process, as long as the resultant overcure remains the same. The Table 2 reveals, however, that the saving in total laser scan time can reach 4 times high if the low resultant overcure of 0.001 inches and low layer thickness of 0.006 inches are used in comparing with the high resultant overcure of 0.004 inches and high layer thickness of 0.012 inches. From these observations, we conclude that the suggested optimal set up mentioned in the previous section can reduce the total laser scan time significantly on the one hand, but it also requires larger amount of total recoating time on the other hand. The resultant total build time (scan + recoating) may increase or decrease when comparing with other setups, which is depending on the specific part's geometry and dimensions in consideration.

7. CONCLUSIONS

A detailed study of the most important five build parameters that affect the quality and accuracy of the final stereolithography parts, namely, the layer thickness, resultant overcure, hatch space, blade gap, and the parts location, was performed in the present investigation. To reduce the large amount of the total number of experiments required, the study employs the Taguchi L27 orthogonal array for setting up the different combination of the respective control factors, each at three different levels. A standard sample was developed which provides a benchmark for comparison of the total of twenty different dimensional, geometry, and surface features. Using the RSM and ANOVA analysis techniques, together with the help of MINITAB software, we were able to identify, among the many thousands measurement data, the factors that are most significant in affecting the quality and accuracy of these twenty representative dimensional and form features, and surface roughness. The analysis results also suggest the best setting of these control factors for each individual feature. For example, to build a square (or rectangular) hole vertically, a low value of resultant overcure (say, 0.001 inches) and medium layer thickness (say, 0.009 inches) must be used to effectively reduce the dimensional error caused by the extra overcure at the down facing layer, and to provide adequate support so that the sagging problems can be eliminated. Aiming at producing the best overall quality parts, the present investigation also proposes an optimal setup of the build parameters for building a general part consisting of a mixture of the various features. Finally, the respective total laser scan time and the corresponding total recoating time were also examined separately for each of the 27 cases studied. It is concluded that the suggested optimal build condition corresponds to the least amount of laser scan time, although the total recoating time may increase due to the smaller layer thickness used. It is noted that, although the conclusions derived from the present investigation are all based on the streolithography parts made from the SLA 250 rapid prototyping machine, the same experimental setup and analysis techniques can be readily applied to different RP technologies, and the corresponding best setting of the various control parameters can be obtained accordingly.

8. REFERENCES CITED

1. Brown, R., Lightman, A., and Wohlers, T., "Need for Expanding the Development and Refinement of Rapid Prototyping Technologies," Newsletter, RPA of SME, Dec., 1994.
2. Jacobs, P., Rapid Prototyping & Manufacturing: Fundamentals of Stereolithography, Published by SME, Dearborn, 1992
3. Jacobs, P., Stereolithography and Other RP&M Technologies, Published by SME, Dearborn, 1996
4. Chen, C. C., and Sullivan, P., "Solving the Mystery-The Problem of Z-Height Inaccuracy of the SLA Parts", Proceedings of the Sixth International Conference on Rapid Prototyping, Dayton, Ohio, 1995, pp.153-174.
5. Gargiulo, E., "Stereolithography Accuracy: A User Study", Proceedings of Stereolithography Users Group Meeting, San Francisco, 1992
6. Vuyyuru, P., Kirschman, C. F. and Fadel, G., "A NURBS-Based Approach for Rapid Product Realization," Web Site of Clemson University, 1995
7. Jayanthi, S., Keef, M., and Gargiulo, E., "Studies in Stereolithography: Influence of Process Parameters on Curl Distortion in PhotoPolymer Models," University of Delaware, 1995.
8. Taguchi, G., "Taguchi Methods, Design of Experiments," Quality Engineering Series, Volume 4, ASI Press 1993
9. Roy, R., A Primer on the Taguchi Method, Van Nostrand Reinhold 1990
10. Ross, P., Taguchi Techniques for Quality Engineering, second edition, McGraw Hill 1996
11. Chen, C. C., and Sullivan, P., "Predicting Total Build Time and the Resultant Cure Depth of the 3D Stereolithography Process", Rapid Prototyping Journal, MCB University Press, England, Vol. 2, Issue 4, 1996, pp.27-40.
12. Myers, R. H. and Montgomery D. C., Response Surface Methodology, John Wiley & Sons, 1995

AN AUTOMATIC REVERSE ENGINEERING APPROACH FOR RAPID PROTOTYPING MANUFACTURING

G. C. I. Lin and L. C. Chen
Centre for Advanced Manufacturing Research
School of Engineering, The University of South Australia
South Australia, Australia

ABSTRACT

Reverse engineering has played an important role in obtaining accurate surface models and shortening product lead time for Rapid Prototyping Manufacturing (RPM). This paper presents an automatic reverse engineering approach to reconstruct Non-Uniformed Rational B-Spline (NURBS) surface models on a coordinate measurement machine (CMM) equipped with a vision system. The vision system is used to detect 3D surface boundaries from physical models to create an initial surface triangular patch. The vertices of the patch are then parameterised into an initial surface model used for further surface refinement digitisation of a touch-triggered probe. The approach aims to integrate surface digitisation and surface modelling into a single adaptive model-estimated process so that free-form surface reconstruction can be implemented accurately and automatically. The problems encountered in traditional reverse engineering processes can be solved simultaneously. As a result, this approach can significantly reduce the traditional product development lead time and obtain satisfactory surface models for RPM.

KEYWORDS

Reverse Engineering, Surface Reconstruction, Automatic Digitisation, NURBS, Rapid Prototyping Manufacturing

1. INTRODUCTION

The design and prototyping of free-form surfaces in the manufacturing industry often involve the use of physical models for initial conceptual and aesthetic design. Usually, product prototyping, performance testing and model modification are needed for obtaining optimal product design. Many reverse engineering applications can be found in industries covering a wide product spectrum. From the automobile industry to aircraft, mould tooling and many others, the physical models with sculptured surfaces need to be reconstructed into CAD models[1, 8]. To satisfy these strong demands, reverse engineering has played a crucial role in obtaining accurate reconstructed surface models and in integrating the whole product design process.

Therefore, an intelligent reverse engineering approach can be very important for a new product development when free-form geometric shapes are applied in the product design. This paper presents an automatic surface reconstruction approach to generate Non-Uniform Rational B-Splines (NURBS) surface models by using a coordinate measurement machine (CMM) equipped with a computer vision system. The proposed approach aims to integrate surface digitisation and surface modelling into a single adaptive model-based process so that free-form surface reconstruction can be implemented accurately and automatically.

The general strategies used to reconstruct a free-form surface model for Rapid Prototyping Manufacturing (RPM) are illustrated in Figure 1. A physical model (clay or wood model) can be created by stylists or obtained from the previous product design cycle (an existing prototype). Using a 3D stereo image detection strategy and a Delaunay Triangulation algorithm[1,2], a surface initial triangulation can be implemented automatically, in order to generate initial surface exploration path(s) for a touch-triggered probe. A Modified Adaptive Model-based Digitising Process (MAMDP) is then used to explore required surface points to create a NURBS surface model which satisfies the user's specified digitising accuracy. The MAMDP provides an automatic surface reconstruction capability to digitise surface points and generate NURBS surface models all in one process. After individual surface entities are created from the MAMDP, they can be further manipulated and assembled by adequate surface manipulation in a computer-aided design (CAD) system, to obtain a complete CAD model. Furthermore, from a surface property evaluation point of view, a 3D surface triangular patch can be also considered as a geometric input to perform surface property analysis (such as finite element analysis) in a computer-aided evaluation (CAE) system. The evaluation result provides the necessary information to modify surface geometry to achieve an optimal surface design. Following this, a complete CAD model can then be used for RPM to generate product prototypes. The intelligent strategies for surface reconstruction, including the Initial Vision-Aided Surface Triangulation Process (IVSTP) and the MAMDP, will be discussed this paper.

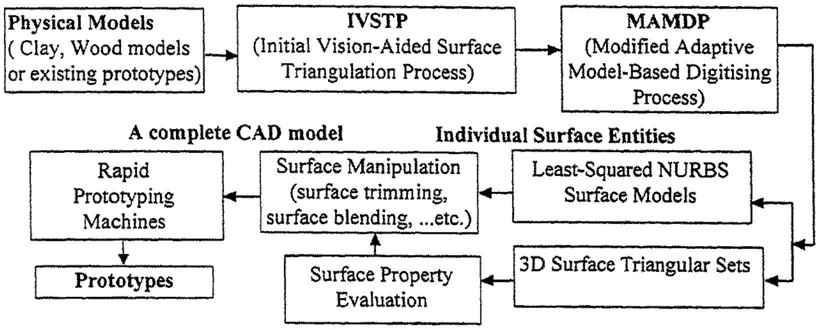


Figure 1 The general strategies of the proposed approach

2. OVERVIEW OF THE PROPOSED APPROACH

The proposed approach integrates two steps (shown in Figure 1), namely the IVSTP and the MAMDP, into a single automatic process of surface reconstruction[1,2]. The IVSTP generates an initial triangular patch by using stereo image detection and constrained Delaunay triangulation. The MAMDP, integrating surface digitisation refinement and surface modelling, is then used to reconstruct surface models and control the digitising accuracy within the required digitising tolerance. These two main steps will be introduced in the following sections.

3. INITIAL VISION-AIDED SURFACE TRIANGULATION PROCESS (IVSTP)

In the IVSTP, a Charge-Coupled Device (CCD) camera is used to rapidly detect the object's position and measure its surface boundary coordinates, to establish geometric information for further automatic surface digitisation of the touch probe.

The flow chart of the IVSTP is shown in Figure 2. In the IVSTP, a vision system detects 3D surface boundaries by using a 3D stereo detection method. Prior to the stereo image detection, multiple images are taken and appropriate image processing methods, such as noise filtering, are applied to reduce image noise to an acceptable level for further processing. A Laplacian edge detection operation is then used to extract 2D surface boundaries effectively. To obtain an accurate boundary measurement, a CCD camera calibration method is also used to establish a camera model which can adequately transform the image coordinates of the surface boundaries to the CMM world coordinates and reduce the image system errors to a minimum. Furthermore, in order to save computing time for further 3D stereo image detection, only characteristic points on the surface boundaries are extracted to represent their original contours by using a data

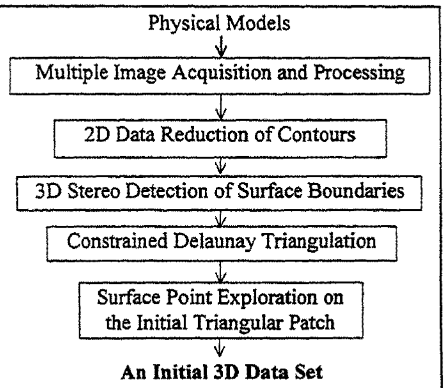


Figure 2 The Initial Vision-Aided Surface Triangulation Process (IVSTP)

reduction strategy developed by the authors[1,2]. Boundary planning is also used to ensure the measuring surface region is capable of being reconstructed by the MAMDP. These extracted points can then be constructed into an initial triangular patch by applying a constrained Delaunay Triangulation. These strategies have been discussed in details in References [1,2,3].

A toy face model (shown in Figure 3) was taken as an example to demonstrate the capability of the IVSTP. Three surface boundaries were automatically detected by the 3D stereo detection method. The characteristic points on the surface boundaries were then triangulated by a constrained Delaunay Triangulation method[4]. As a result, a 3D initial triangular surface patch can be generated and used for probe exploration paths in the following MAMDP.

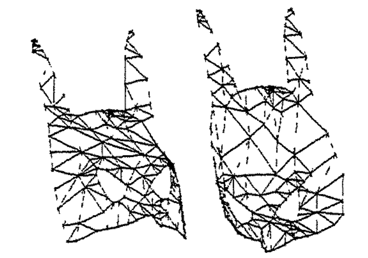


Figure 3 The 3D initial triangular patch of a reconstructed toy model (the left picture is the right ISO view and the right picture is the left ISO view)

4. MODIFIED ADAPTIVE MODEL-BASED DIGITISING PROCESS (MAMDP)

An adaptive digitising process, called the Adaptive Model-based Digitised Process (AMDP), was developed to digitise a sufficient number of surface points and reconstruct least squares B-spline surface in one single process[5,6]. A least-squares bi-cubic B-spline surface model was applied as the evaluated model in this adaptive process, to evaluate the digitising accuracy for each subdivided triangular patch and to calculate the new estimated surface coordinates for the next digitising cycle. The application examples used in their research have shown that the digitising data can be controlled to meet the user-specified digitising tolerance and least squares bi-cubic B-spline surface models which are smooth and where fitted accuracy can be obtained. However, in order to provide higher flexibility in reverse engineering a large variety of analytical entities and free-form shapes, NURBS surfaces are applied in the MAMDP as the evaluated model. NURBS has been well known for its many advantages in computer aided geometric modelling (CAGD). By manipulating the control points as well as weights, NURBS provides the flexibility to design a large variety of geometric shapes. The format of NURBS is a general expression under which the spline, Bezier, and conic format exchange easily. Therefore, a least squares NURBS surface modelling technique is applied in this digitising process, not only to optimise weights and control points for obtaining the best fitted surface models but also to generate CAD-compatible surface models for further manufacturing applications.

4.1 Mathematical Modelling of Free-Form Curves and Surfaces From Discrete Digitised Points With NURBS

A NURBS surface[7] is the rational generalisation of the tensor-product non-rational B-spline surface and is defined as follows:

$$S(u,v) = \frac{\sum_{i=0}^{n_u}\sum_{j=0}^{n_v} w_{i,j} P_{i,j} B_{u,i}(u) B_{v,j}(v)}{\sum_{i=0}^{n_u}\sum_{j=0}^{n_v} w_{i,j} B_{u,i}(u) B_{v,j}(v)} \quad (1)$$

where $\left\{\left\{w_{i,j}\right\}_{j=1}^{n_v}\right\}_{i=1}^{n_u}$: the related weights

$n = n_u \times n_v$: the control points

$\left\{B_{u,i}(u)\right\}_{i=1}^{n_u}$, $\left\{B_{v,j}(v)\right\}_{j=1}^{n_v}$: the normalised B-splines with order k_u , k_v in the u and v directions respectively. The knots are defined over $\left\{t_{ui}\right\}_{i=1}^{n_u+k_u}$ and $\left\{t_{vi}\right\}_{i=1}^{n_v+k_v}$

$P_{i,j}$: the control points

Most NURBS surface models can be generated by directly setting adequate knots and manipulating weights and control points. However, in the case of reverse engineering, many unknown parameters, such as control points and weights, need to be identified from digitising points. Searching optimal parameters for those unknown parameters used in the surface modelling is required. Moreover, measured errors due to the limited accuracy of the digitising devices normally introduce instabilities to the reconstructed surface when interpolation techniques are applied to fit through digitised data. In the presence of random errors in digitised points, the fitted curve or surface may exhibit severe distortion and instabilities[8]. This explains why least squares approximation techniques are popular in reverse engineering to avoid the problem, as they can generate smooth surface models having minimum fitted deviation between the surface and the digitised data.

A wide variety of research has been undertaken to reconstruct curves or surfaces from the measured data[8, 9, 10]. Piegl[9] employed a constructive approach for curve fitting. Piecewise Rational Bezier Curves are first constructed from measured points and then connected together to form a NURBS curve. However, this approach does not guarantee the best fitting NURBS. In recent years, there has been successful research in non-linear approaches. The major disadvantage of these methods is that a long computation time is required for the minimisation and the difficulties for managing positive weights are not easy to overcome. Saker and Menq[10] applied nonlinear least-squares minimisation to find a smooth parametric surface approximation. The objective function minimised is the explicit error expression for the sum of the squares of error values at the data points. This means that the difference between the measured data points and the corresponding points on the fitted surface is minimised in the least-squares sense, and no refitting of the surface is required. However, this approach is still limited to fit regularly distributed points. Ma[11] proposed a two-step linear approach and the concept of base surface to optimise the weights and control points for parameterising randomly distributed points with NURBS. This method can be applied to the interpolation and fitting of general NURBS curves and surfaces from discrete points. All weights of the control points were first identified by the Singular Value Decomposition (SVD) technique. The control points were then further solved in the least squares sense by applying Householder transformations. The fitting results were found to be more accurate than those created from the surface modelling using uniform weights.

In the MAMDP, there are two basic considerations in the surface model estimation. One is that all digitised points are distributed in a triangular patch in which the points are not regularly distributed in a rectangular order. The surface modelling method applied in the adaptive process should be capable of parameterising the data with this kind of data format. Another consideration is that the time spent in searching for the optimal parameters should be controlled within an acceptable range. Non-linear least squares optimisation methods for surface fitting may still have difficulties in meeting this requirement. Therefore, Ma's method[11], satisfying these two requirements, was then applied to integrate with the adaptive model-based digitising process. The strategy for the least squares fitting from randomly distributed points is discussed as follows.

The base surface is generated from the surface boundaries in the form of either points or curves. This base surface is the first approximation of the final fitted surface. It provides the tool to locally parameterise the digitised points in the u and v directions. Following this, the strategy for least squares fitting can be applied to any kind of features when the mathematical model of the geometric feature can be expressed in a set of equations. The least squares fitting can be defined as Equation (2).

$$\underset{X}{Minimize} \ \sum F(X) = \sum_{i=1}^{m} \left\| f_i(X) \right\|_2^2 \qquad (2)$$

Where $f(X) = 0$: the observation of the mathematical model

X : the vector containing the definition parameters with which the underlying surface is uniquely defined (such as the control points and the weights).

The Equation (2) can also be expressed as Equation (3) for the NURBS surface fitting.

$$\underset{X}{Minimize}\ S = (B \cdot X - \overline{X})^T \cdot (B \cdot X - \overline{X}) \tag{3}$$

Where B : the collection of the n normalised B-splines

X : the collection of the coordinates of the control points in 4D homogeneous space

The detailed strategies of the two-step linear approach was described in Reference 11.

4.2 Digitising Strategies

The initial surface triangular patch, generated in the IVSTP process, initialises the modified adaptive digitising process. As shown in Figure 4, the base surface can be first generated by parameterising the surface boundaries in the form of curves when the number of control points and the order of B-spline are adequately set. The digitised points within the surface boundaries are then parameterised in the u and v directions. Hence, the first approximate NURBS surface model can be generated. Following this, the surface model is used to calculate the new exploration points at the approximate midpoints between the vertices of the initial triangular patch. The CMM then automatically digitises these exploration targets. In the next step, this adaptive approach evaluates the digitising accuracy by calculating the deviations between the target points and the digitised points. As a result, the calculated deviations can be used to evaluate whether or not further exploration will be undertaken in each subdivided triangular area. New exploration targets will only be generated at the estimated midpoints between vertices of those triangular patches where deviations are higher than the user-specified tolerance. This iterative digitising loop will continue until all new digitised deviations are within the specified tolerance.

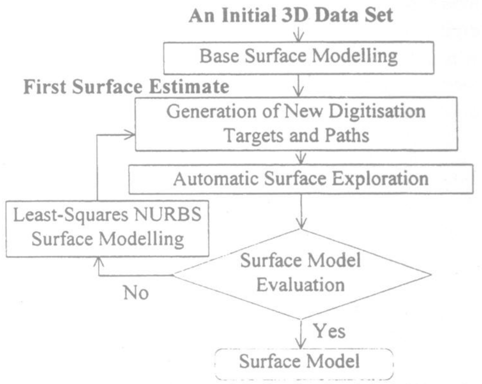


Figure 4 The Modified Adaptive Model-Based Digitising Process (**MAMDP**)

5. INDUSTRIAL APPLICATION

One industrial example was investigated to test the feasibility of the proposed approach. Reverse Engineering a turbine blade (shown in Figure 5) is important when its surface model needs to be updated in its design process. Therefore, an accurate surface model of the turbine blade has to be reconstructed from the physical model after the required modification has been carried out on the model.

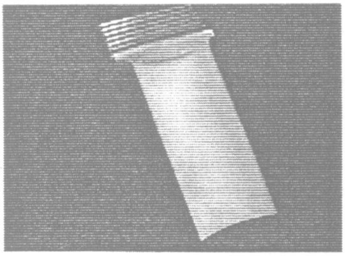


Figure 5 The physical model of a turbine blade

The proposed approach has been implemented on a *Brown&Shape* CMM with a *Renishow* touch trigger probe and a vision system from the *Matrox* company. By using the IVSTP, the initial surface triangular patch (shown in Figure 6) was generated. This patch provided the probe's exploration paths. Following this, a surface triangular patch (shown in Figure 7) and a surface model (shown in Figure 8) were generated by using the MAMDP. As shown in Figure 7, it is obvious that the distribution of the digitised points follows the local surface-curvature trend. The orders of B-spline was set at 4 and the number of control points at 12 x 12 which was less than $(2m + 1) / 3$, where m (= 623) represents the number of digitised points. The user-specified digitising tolerance was set at 0.25 mm. As a result of the surface reconstruction, the maximum surface fitted deviation was 0.1712 mm and the average deviation 0.005 mm. The processing time spent for reconstructing the surface models was 46 minutes. The results show that the time required in surface model reconstruction can be significantly reduced from one or two days, which are needed in a traditional reverse engineering process, to less than an hour. The accuracy of the reconstructed surface models was also controlled in a good range.

Figure 6 The 3D initial triangular patch of a turbine blade (the left picture is the right ISO view and the right picture is the left ISO view)

Figure 7 The surface triangular patch of the turbine blade (the left picture is the right ISO view and the right picture is the left ISO view)

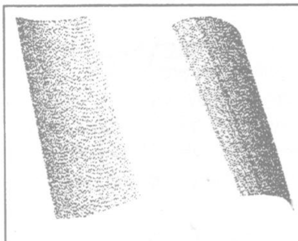


Figure 8 The surface model of the turbine blade (the left picture is the right ISO view and the right picture is the left ISO view)

6. CONCLUSION

The proposed approach provides an effective method to automate the digitisation process of free-form surface reconstruction and to obtain a least-squares NURBS B-spline surface model with controlled digitising precision. The greatest advantage is to integrate two main steps in reverse engineering, namely surface digitisation and surface modelling, into a single intelligent reconstruction process. This strength significantly enhances the measuring speed of a CMM and also maintains high accuracy measurement for reverse engineering applications. Obviously, from checking the surface triangular patch in the example, only the surface characteristic points are captured for the surface modelling and no redundant surface point is digitised. This avoids difficulties in dealing with large amounts of redundant points in the traditional reverse engineering approaches. Moreover, the digitised result can be guaranteed to meet the user-specified digitising accuracy. The CMM equipped with a touch-triggered probe also provides the accurate digitised measurement. This avoids difficulties in dealing with the digitised points having unacceptable measured errors. All of these advantages demonstrate that the approach provides an effective method in solving the existing problems encountered in reverse engineering processes. Future development will focus not only on the mathematical evaluation of the adaptive digitising process but also on system integration, to achieve a higher level of automation.

7. REFERENCES

1 Lin, G. C.I. and Chen, L.C., "A Vision-Aided Reverse Engineering Approach to Reconstruct Free-Form Surfaces", *Proceeding of CAD/CAM Robotics and Factories of the Future*, pp 854-859, London, England, August 14-16, 1996

2 Lin, G. C. I. and Chen, L.C., "An Intelligent Surface Reconstruction Approach for Rapid Prototyping Manufacturing", *The Fourth International Conference on Control, Automation, Robotics and Vision* (ICARCV'96), pp. 43-47, Singapore, 3-6 December, 1996.

3 Chen, L.C. and Lin, G. C.I., "A Vision-Aided Reverse Engineering Approach to Reconstructing Free-Form Surfaces", Submitted to *Journal of Robotics and Computer Integrated Manufacturing*, December, 1996.

4 Ruppert J, "A Delaunay refinement algorithm for quality 2-dimensional mesh generation", *Journal of Algorithms*, **18**, pp. 548-585, 1995.

5 Lin, G. C.I. and Chen, L.C., "An Integrated Reverse Engineering Approach to Reconstruct Free-form Surfaces in Coordinate Measurement Machines", *The Fourth International Conference on Automation Technology, Automation '96*, pp. 239-246, Mechanical Industry Research Laboratories / ITRI, Hsinchu, Taiwan, R.O.C., July 8-11, 1996.

6 Chen, L.C. and Lin, G. C.I., "An Integrated Reverse Engineering Approach to Reconstructing Free-form Surfaces", (accepted for publication in the journal of Computer Integrated Manufacturing Systems), 1997

7 Tiller, W., "Rational B-Splines for curve and surface representation", *IEEE CG&A*, 0272-1716/83/0900.

8 Ma, W., NURBS-based CAD Modelling From Measured Points of Physical Models, PhD thesis, Katholieke Universiteit of Leuven, 1994.

9 Piegle, L., "Interactive data interpolation by rational Bezier curves", *IEEE CGEA*, April 1987, pp 45-61.

10 Sarkar, B. and Menq, C.H., "Smooth-surface approximation and reverse engineering", *Computer Aided Design*, Vol. 23, No. 9, Nov. 1991, pp 623-628.

11 Ma, W. and Kruth, J.P., "Mathematical modelling of free-form curves and surfaces from discrete points with NURBS", in P.J. Laurent, A. Le Mehaute, L.L. Schumaker, *Curves and Surfaces in Geometric Design*, pp 319-326, A. K. Peters Ltd, Wellesley, MA, (1994).

REVERSE ENGINEERING BASED ON MULTI AXIS DIGITIZED DATA

Chr. Schoene and J. Hoffmann
Institut of Production engineering
The University of Technology Dresden
Germany

ABSTRACT

This paper is focussed on improved methods to measure workpiece shape (geometry) via digitizing or scanning. These methods as well as the scope of Rapid Product Development became an essential supposition for Reverse Engineering in the last two years. Surface feedback (to get a analytically described surface) based on digitized data needs a very long time. The paper shows a alternativ way, the triangulation method based on multi axis digitized data.

KEYWORDS

Digitizing, Reverse engineering

1. MACHINE TOOLS TO BE USED FOR MULTI AXES DIGITIZING

Commercial systems of 5 axis digitizing equipment have been known for the previous two years /1,2/. Hereby, rotary machine axes are used for workpiece or probe/ laser positioning to record the shape of a part or model entirely from all sides. As a disadvantage of those systems, specific characteristics and extended options for further processing of digitized points as enabled by rotary machine axes are supported insufficiently, only.

At the beginning, practice investigations of multi axis digitizing were carried out on a HURON type milling machine fitted with a Heidenhain digitizing system /3/. On the milling machine, horizontal and vertical spindle are available. Axis location is elucidated in Figure 1. To realize an additional rotary machine axis, the machine can be extended by an indexing attachment with rotary table. Further practical tests were carried out on a 5-axis milling machine MAHO800C with laser scansatation.

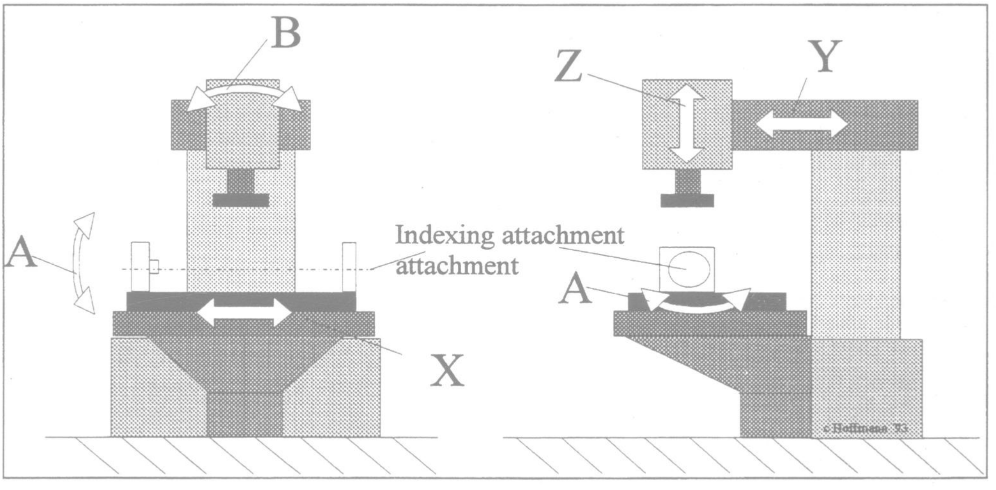


Fig. 1: HURON type milling machine fitted with a Heidenhain digitizing system; axis assignment

2. PROCESSING OF MULTI AXIS DIGITIZED DATA

To mill directly from digitized points is regarded as the nowadays state of the art. Mostly, corresponding data handling is directly done on the shopfloor. However, preparation of point clouds (scatter diagrams) for generation of curves and surfaces as well as triangulation (that is shaping a net structure of triangular surfaces) demand for significant computational and also manual expenditures. Following theoretic assumptions and our works on digitizing, for follow-up 3D-CAD modelling, workpieces are classified into 3 families. Thereby, the parts are grouped according to the geometric aspects following:

1. Workpieces incl. surfaces represented by functional relations (e.g. turbine blades, compressor impellers)
2. Parts with designer surfaces such as car bodies and consumer goods
3. Pieces of strongly structured surfaces like medical items and ground patterns

Digitizing parameters necessary for the typical part families mentioned above are specified in Figure 2:

Part family	Distance between digitizing paths	Point distance on digitizing path	Rotary axis necessary
1	select big distance, for individually selected paths, only	select big distance	absolutely necessary
2	select small distance according to model accuracy	select small distance	sometimes necessary
3	select very small distance	select very small distance	sometimes necessary

Fig. 2: Digitizing parameters depending on three types of part families

3. DIGITIZING FOR FOLLOW UP PROCESSES OF RAPID PROTOTYPING (RPD) SUPOSITIONS

3D part modeling stands for an essential supposition of part manufacturing by means of RPD methods. Commercial RPD systems for RPD process modeling additionally offer slicing operations based on STL formatted CAD data /4/. Commonly used CAD systems as mentioned by /5/ optionally provide STL interfaces. All generated STL files are based on surfaces analytically described in any case. These surfaces are provided in the VDA or IGES formats.

When digitizing data for follow up RPD processes, an entire representation of workpiece geometry is absolutely necessary. If parts will be digitized incompletely, only, there must be constructed auxiliary surfaces to round or to close a pattern. The resting activities to process digitized data for workpieces with polynomial surfaces can be completely taken from surface feedback operations which are identical.

As evidenced by the authors, there could be implemented the entire process chain following:

1 digitizing
2. preparation of point cloud, select characteristically points
3. generation of curves - generation of polynomial surfaces
4. triangulation upon polynomial surfaces
5. slicing
6. RPD manufacturing.

This entire process could be demonstrated for a turbine blade (see Figure 3 and /6,7/).

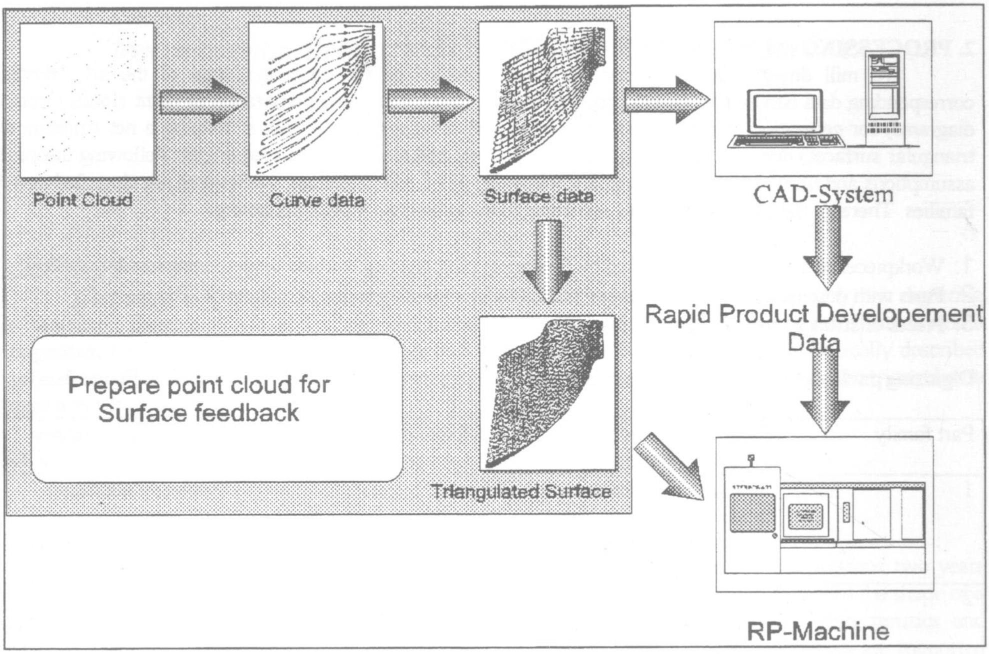


Fig. 3: Process chain: Digitizing - generation of curves and surfaces -triangulation -slicing - RPD, for a turbine blade

Having in mind processing of digitized data for RPD manufacturing, the procedure carried out until now could be done much quicker. As known from literature /8/, there are available software systems for triangulation immediately based on ordered point clouds which are free of undercuts. Latter are point clouds which can be projected to one layer unambiguously. Triangulation data could be used for slicing as a base of follow up RPD manufacturing. Triangulation of point clouds is provided by CAD systems or systems for surface feedback like software SURFACER /8/. Figure 4 shows an example for triangulation of an surface Nevertheless, the authors don't know any applications to use these data for slicing in order to manufacture a part via RPD. As evidenced by our experience with STL data files, unordered point clouds with undercuts often result in faulty STL files. In the case of ordered point clouds without any undercut, it is true, that there are mostly achieved sufficient results. However, in the STL file, gaps, doubling and malorientations occur. Futhermore, these data are not suitable for slicing: These patterns are often not closed caused by lacking side and base surfaces. Subsequently, therefrom it is necessary to generate interactive tools for handling STL data files. This subject is under investigation now. Additionally, the authors carried out investigations to find out an intelligent algorithm for triangulation upon point clouds. As an objective, this algorithm should be able to

a) process point clouds even with undercuts
b) handle unordered point clouds.

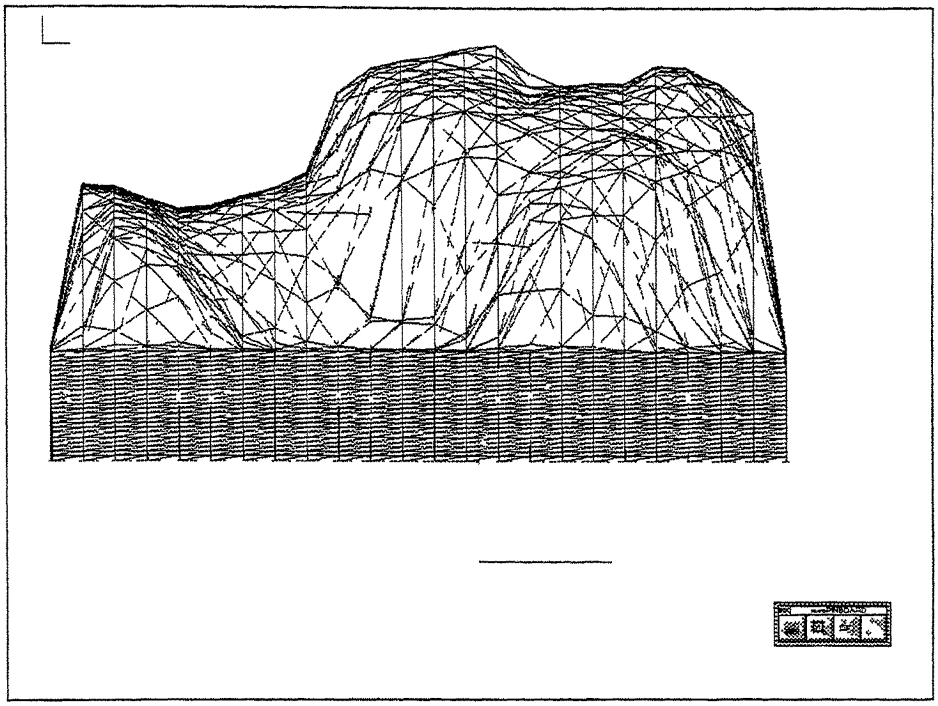


Fig. 4: With software SURFACER /8/ triangulated surface of a sculpture

The algorithms described by references are not able to cope with these suppositions mentioned. As a consequence, the variety of several point cloud characteristics demands for several processing strategies. Triangulation upon unordered point clouds with undercuts cannot be carried out entirely automatically. For that reason, the authors oriented to an iterative algorithm named „Triangulation by local search and completion", on the one hand. On the other hand, we thought about suitable initial parameters to be preset before the point cloud's triangulation /9/. These initial parameters depend on the point cloud's characteristics (see figure 5).

Parameter	Explanation	Destination
DMAX	permitted maximum edge length of a triangle	Thereby it is avoided, that undesireable net holes are filled automatically. This parameter also defines the range of points to be considered for calculation. Thus, the running speed of this algorithm is essentially influenced.
WAMAX	maximum internal angle when generating a triangle	Interior angle WAMAX when generating a triangle by adding a point has to be less than the defined value.
WCMAX	maximum interior angle between neighboured outer edges	If the angle between neighboured outer edges exceeds the WCMAX value, then a triangle may be generated by pasting a curve.
WNMAX	maximum angle between the triangles' surface normals	This value defines the permitted tilting of triangles one against the other.

Fig. 5 : Initializing parameters for the triangulation algorithm /9/

In figure 6 is illustrated a example of a triangulated surface with the algorthm „Triangulation by local search and comletion".

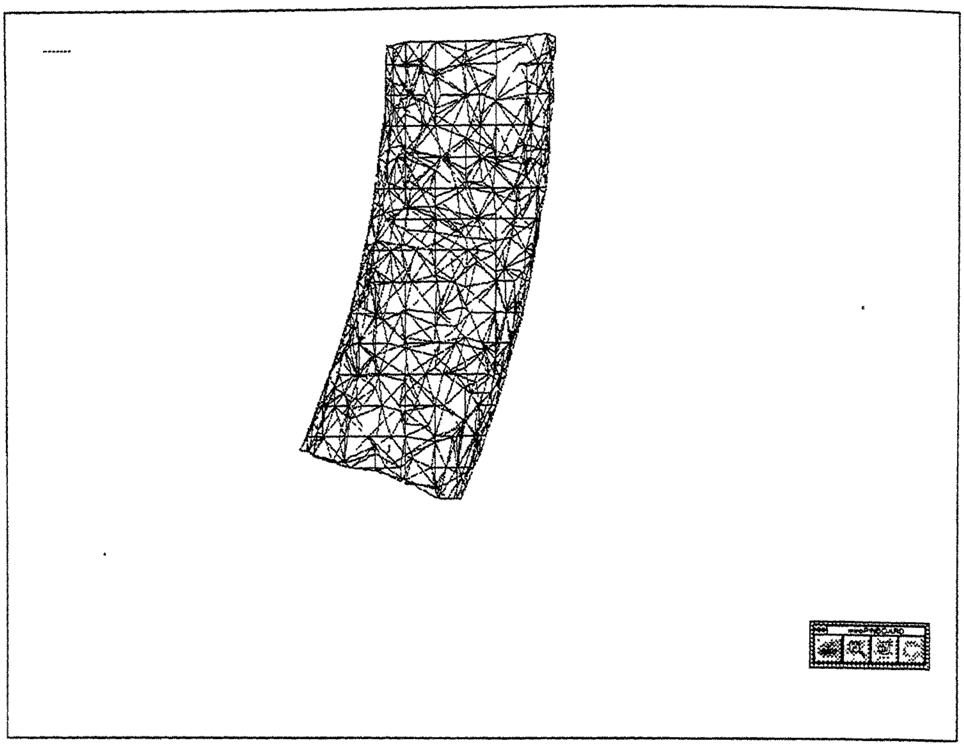


Fig 6: With triangulation algorithm /9/ generated file of a part of a steering wheel

As a result of research, an algorithm for triangulation of unordered point clouds with undercuts was generated. Ongoing works are directed to analyze the quality of triangulized surfaces. Quality comparison is focussed on these two approaches based either on polynomial surfaces or on point clouds.

4. REFERENCES

1 Just,G., Sonntag,A.: Erfahrungen und Besonderheiten beim 3D Vermessen von Designmustern, Internationale Konferenz „Internationale Produktionssysteme", Solid Freeform Manufacturing", Dresden, September 1995

2 Massen, R., u.a. Optisches 3D Digitalisieren mit ungenauen Achsen, Internationale Konferenz „Intelligente Produktionssysteme, Solid Freeform Manufactturing" Dresden, September 1995

3 Bedienhandbuch zum Digitalisiersystem LEMOINE auf HURON 4 Achsen Fräsmaschine, 1994

4 Bedienungsanleitung STEREOS 600, EOS GmbH

5 Produktbeschreibung, Digitalisierdatenverarbeitung mit Strim 100, 1994

6 Fichtner, D., Schöne, C., Hoffmann, J.: Alternative Prozeßkette beschleunigt Produktentwicklung, zur Verföffentlichung eingereicht bei VDIZ, 199,5

7 Fichtner, D., Hoffmann, J.: Process Chains for Rapid Prototyping, Vortrag zur INCOM'95, 8th Symposium on Information Control Problems in Manufacturing, Peking, 11.-13. Oktober, 1995.

8 Produktbeschreibung SURFACER von 3D Imageware, USA 1994

9 Maukisch. M.: Erarbeitung und Untersuchung von Triangulationsalgorithmen auf der Basis von Digitalisierpunktwolken, Diplomarbeit, TU Dresden, Institut für Produktionstechnik, 1996

RANGE-IMAGE MACHINING (RIM) - A TOOL FOR THE PRODUCTION OF HIGH ACCURACY RADIOTHERAPY MASKS

Ralph W. L. Ip* and **Felix T. S. Chan****

*Department of Manufacturing Engineering, City University of Hong Kong, Hong Kong
**Department of Industrial and Manufacturing Systems Engineering,
The University of Hong Kong, Hong Kong

ABSTRACT

Thousands of people suffer cancer disease every year. The cancer cells may be found on the face of the patients. The possible medical treatment is to use radiotherapy to remove the cancer cells. The treatment need to be performed every day and the whole treatment period may be extended to a few months. In order to provide a base to place the radiotherapy instrument at a precise position on the patient's face, a mask need to be made. The traditional mask is made of gypsum, the patient needs to stand the bad smell and will feel very uncomforted during the whole mask making process. The new investigated Range-Image Machining (RIM) approach is an alternative solution to produce the mask, it can provide a better environment to the patients because the gypsum mask need not be made. The RIM approach is built on the concept of reverse engineering, which consists of the techniques of non-contact co-ordinate measurement, surface modelling and computer numerically controlled (CNC) machining. In the approach, face profile of the patients is extracted by scanning, then bi-cubic B-spline surface model is used to reconstruct the profile and realistically reproducing it on a CNC machining centre. The range data need to be pre-filtered before modelling. The refined data are then used to reconstruct the face profile in the surface remodelling process. In making the mask master, a ball-nosed cutter is driven forward and backward on a CNC machining centre to trace the defined cutter paths. The aim of the paper is to present the entire principle of the RIM approach in mask making for Nasopharyngeal Carcinoma (NPC) patients, the details of each process as well as the surface reconstruction algorithm will also be addressed.

KEYWORDS

Reverse Engineering, Laser Scanning, Surface modelling, CNC machining

1. INTRODUCTION

In the past, mask is mainly used by actors and actress during performance. Since the industrial revolution, mask is needed for the purposes of protection when the operators work in a skin harmful environment, such as chemical manufacturing plants and steel refining factories. Recently, mask is employed as a tool in radiotherapy. This new idea of using mask in the medical technology was initiated in the Pamela Youde Nethersole Eastern Hospital (PYNEH) in Hong Kong. Statistical results from the Hong Kong Hospital Authority stated that cancer was one of the top murderers over the past ten years in Hong Kong. The cancer cells may be found anywhere on a patient, however, Nasopharyngeal Carcinoma (NPC) commonly appears on human head. The patients usually suffer psychological damage more than physical pain, the situation will be more serious for the cases of facial deformation.

Traditionally, radiotherapy is a positive treatment method for cancer. For a NPC patient, the cancer cells are normally found in the deep part of the skull. A gun shape radiotherapy instrument is needed for the treatment. The treatment is normally to be carried out every day and the entire treatment period may be extended to two or three months. In order to give a good treatment, the instrument should be accurately placed at the specified positions on the patient's face. Before the development of radiotherapy mask, the radiotherapists will mark lines and points on the patient's face for the aim of positioning the instrument. This will give extra psychological damage for the patients because they need to wear the marking for the entire treatment period.

Radiotherapy mask making is a new and accurate strategy to assist the positioning of the instrument on the patient's face. The mask is tailor-made for each patient, because it must fit for the whole face profile without any excessive gaps. Reverse engineering [1] is adopted in the mask manufacturing process. The mask is produced through a serious of measurement, modelling and manufacturing process with the aid of computing technology. The aim of the research paper is to report the newly invented reverse engineering technique, namely Range-Image Machining (RIM), for the manufacturing of radiotherapy masks. The entire mask manufacturing process as well as the face profile acquisition and reconstruction are detail described. A clinical test is also presented which was a substantial trial to validate the technique.

2. RANGE-IMAGE MACHINING - A REVERSE ENGINEERING APPROACH

The RIM approach is built on the concept of reverse engineering. It was developed in parallel with a similar approach: Shaded-Image Machining (SIM). The approach was invented by Ip [2, 3] in the University of Birmingham, it is a new shape recovery strategy for free-form features. The major different of the two approaches is the former one to use range data to reconstruct the surface profile, however, the latter approach employs photometric stereo techniques and the shape from shading model [4, 5] to rebuild the surface geometry from a set of captured half-tone images through computer vision techniques.

In the model of RIM, two major processes are needed to reconstruct a surface profile from a master, i.e. surface feature information acquisition and surface geometry remodelling. The results of the remodelled surface are used to produce the masters for the duplication of plastic radiotherapy masks.

2.1 Surface feature information acquisition

Two common types of surface feature information acquisition methods are available, they are direct contact and non-contact co-ordinate measurement. Direct contact measurement is normally carried out on a co-ordinate measurement machine (CMM). Some research reports [6, 7] have addressed the principles of co-ordinate measurement, in addition the techniques in direct contact co-ordinate measurement for nominal geometry and free-form features can be found from British Standard [8] and research literature [9, 10]. Ip and Loftus [11] have reported an accurate strategy to assess the geometrical co-ordinate of a machined free-form surface.

The basic principle of non-contact co-ordinate measurement is similar to direct contact measurement, however, the technique in feature information acquisition is different. The RIM approach is to use the non-contact co-ordinate measurement method to collect the feature information from a master surface. The reason lies in the fact that the master surface, e.g. a human face, can easily be deformed by the measuring probe in direct contact measurement. Discrepancy is introduced to the remodelled surfaces when these inaccurate data are employed.

Laser scanning is used commonly in non-contact co-ordinate measurement. Point scanning and line scanning are different strategies in designing a laser scanner. Actually, the acquired information either from a point or a line scanner is a group of range data. The most simplest format of the range data is a set of ordered co-ordinate data. They can be recorded under Cartesian co-ordinate or Polar co-ordinate framework as illustrated in Figures 1a and 1b respectively.

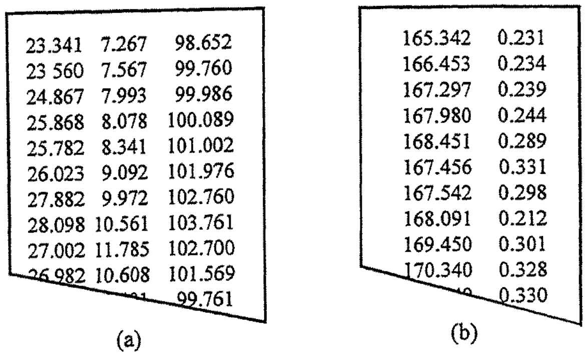


Figure 1 Two types of range data format: (a) Cartesian co-ordinate; (b) Polar co-ordinate.

2.2 Surface geometry remodelling

The acquired range data from a master surface by laser scanning is just a group of ordered co-ordinate data as shown in Figure 1. There is no geometrical relationship is given from the data. The second process in the RIM, it also is an important step, is to rebuild a surface from the range data using geometrical surface models. Piegl and Tiller [12] stated that the surface models use mathematical methods to present a surface in a geometrical space, the relative information of the modelled surfaces, e.g. tangency, curvature can also be interrogated through analytical equations.

In the RIM approach, bi-cubic B-spline surface model [13, 14] is selected as a tool to rebuild the surface profile features from the acquired range data. Firstly, the advantage of using B-spline model is it can provide a second order parametric continuity feature on a composite surface. Secondly, less control polygon nets need to be defined in fitting a surface in the rebuild process. In other words, the calculation in surface fitting can be faster than the other surface models, such as Bezier.

The acquired range data are normally to be refined before to carry out surface remodelling process. The data with abnormal deviation and the oscillating data are all filtered, see Figure 2.

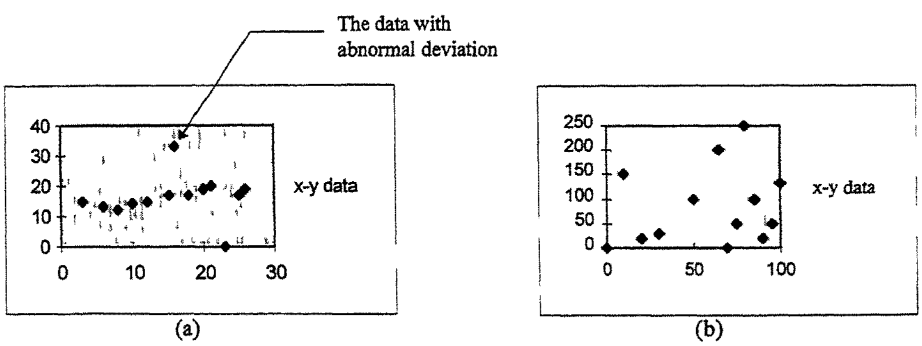


Figure 2 Two common type of range data need to be filtered: (a) abnormal deviation; (b) oscillating data.

The filtered data are then put into the bi-cubic B-spline surface equation to determine the control polygon nets. A surface fitting tolerance limit ε and the maximum number of polygons V_m

need to be defined before the surface remodelling process. The control polygon nets are used to determine B-spline surface patches to fit the range data. In mathematical expression, the control polygon net is written as:

$$[V] = [M(u)]^{-1} [R] [N(v)]^{-1} \qquad (1)$$

where $[V]=\begin{bmatrix} B_{0\,0} & B_{0\,1} & B_{0\,2} & B_{0,3} \\ B_{1\,0} & B_{1\,1} & B_{1\,2} & B_{1\,3} \\ B_{2\,0} & B_{2\,1} & B_{2\,2} & B_{2\,3} \\ B_{3\,0} & B_{3\,1} & B_{3\,2} & B_{3\,3} \end{bmatrix}$, $[R]=\begin{bmatrix} r_{0\,0} & r_{0,1} & r_{0,2} & r_{0\,3} \\ r_{1\,0} & r_{1,1} & r_{1\,2} & r_{1\,3} \\ r_{2\,0} & r_{2,1} & r_{2\,2} & r_{2,3} \\ r_{3\,0} & r_{3\,1} & r_{3,2} & r_{3,3} \end{bmatrix}$,

$$[M(u)] = \begin{bmatrix} u^3 & u^2 & u & 1 \end{bmatrix} \frac{1}{6} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{bmatrix} \quad \text{and} \quad [N(v)] = \frac{1}{6} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 0 & 4 \\ -3 & 3 & 3 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ v \\ v^2 \\ v^3 \end{bmatrix}.$$

In equation (1), $[V]$ is the control polygon net, $[R]$ is the filtered range data, $[M(u)]$ and $[N(v)]$ are the basis functions of the B-spline surface equation. In theory, a single B-spline surface patch is determine from a group of sixteen range data. A new adjacent patch can be created in case of four more range data are added into equation (1). Exactly fitting the range data by a bi-cubic B-spline surface is very time consuming, a practical approach is to use the surface model to approximate the range data within the pre-defined tolerance limit ε. In fact, more iteration steps are needed when a closed tolerance limit is assigned. The number of polygon net can also be controlled to prevent an excessive iteration of a tight tolerance surface.

2.3 Manufacture the reconstructed surface

Producing the remodelled surfaces is a CAD/CAM integrated manufacturing task. Computer numerically controlled (CNC) machining techniques are employed in the RIM approach, however, a number of prototype production techniques are available, such as stereolithography (STL) and Laminated Object Manufacturing (LOM). In CNC machining, the geometrical information of the remodelled surfaces are used to calculate the cutter paths. A hemispherical head end mill is driven by a series of NC codes, which are determined from the surface geometry data, to perform the cutting process.

3. RADIOTHERAPY MASK MAKING

A joint investigation of producing a radiotherapy mask with the PYNEH was carried out in the Department of Manufacturing Engineering, City University of Hong Kong. In this investigation, the face profile of a NPC patient was scanned by a single point laser scanner as illustrated in Figure 3. The largest scanning depth of field is 400 mm and the scanning track is a 180 degrees circular path. The filtered acquired range data from the patient's face are shown in Figure 4, and the remodelled face profile is illustrated in Figure 5. A hard-wood block was used as the cutting material, and the entire cutting operation was carried out on a 4-axis horizontal machining centre. The reproduced face profile on the wooden block was then used as a master to produce the radiotherapy mask on a vacuum forming machine. Figure 6 shows the finished plastic radiotherapy mask can accurately fit the face profile of the patient.

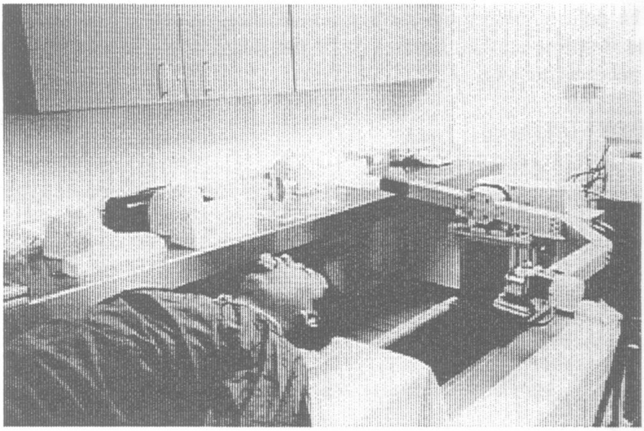


Figure 3 The point laser scanner used for face profile measurement.

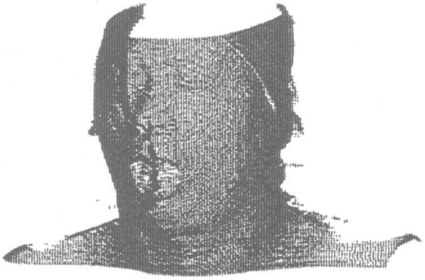


Figure 4 The filtered acquired range data.

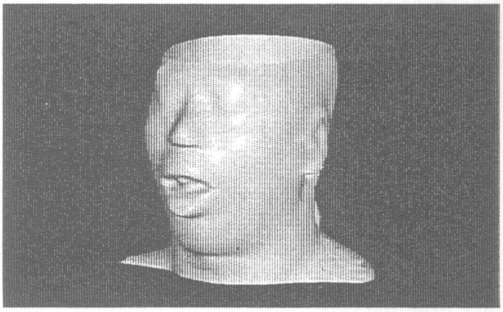


Figure 5 The remodelled face profile.

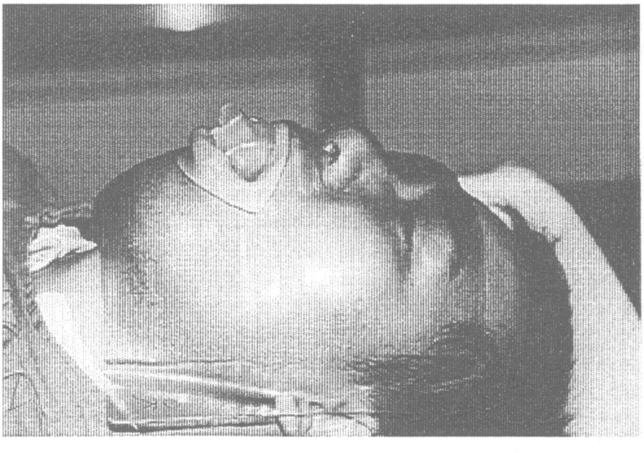


Figure 6 The plastic radiotherapy mask was tested by the patient.

4. DISCUSSION AND CONCLUSION

The results of the joint investigation substantially demonstrated the feasibility of RIM in making radiotherapy masks. Using the RIM approach can be an accurate strategy in acquisition and reproducing the highly complex shape surfaces, e.g. human face. In fact, the geometrical precision of the masks will depend upon the accuracy of the scanner, and the chosen tolerance limit in the surface reconstruction process. The scanner accuracy and tolerance limit used in the investigation were 0.1 mm and 0.2 mm respectively.

In conclusion, a mask making technology for radiotherapy was developed which based upon the principle of reverse engineering. The RIM approach consists of three major process in producing a radiotherapy mask: surface feature information acquisition, surface geometry remodelling and mask profile manufacturing. Laser scanning technology is used to acquired the range data, B-spline surface model is employed to rebuild the surface profile and the mask master is manufactured on CNC machines.

5. ACKNOWLEDGEMENTS

The authors wish to thank the Hospital Authority for financial support in conducting the joint research project between PYNEH and City University of Hong Kong. Technical support from the mould centre in the Radiotherapy and Oncology Department of PYNEH is gratefully acknowledge. Dr Ip also wishes to thank his colleagues Mr. H.Y. Wong and Mr. C.C. Leung for their helpful support with respect to the research.

6. REFERENCES

1 Bradley, C., Vickers, G.W and Milroy, M., "Reverse engineering of quadric surfaces employing three-dimensional laser scanning", Proceedings of IMechE, Part B, Journal of Engineering Manufacture, Vol 208, pp 21 - 28, 1994.

2 Ip, R.W L., Enhanced procedures to model, machine and measure free-form surfaces using analytical and artificial intelligence techniques, Ph D. Thesis, University of Birmingham, U.K. 1993.

3 Ip, R.W.L., "Shaded Image Machining (SIM)· A complex curved surface reproduction tool", 1 st International Machining & Grinding Conference, Michigan, US, pp 721 - 734, 12 - 14 September 1995.

4 Horn, B.K.P., "Understanding image intensities", Artificial Intelligence, Vol. 8, pp 201 - 231, 1977

5 Ikeuchi, K and Horn, B.K.P., "Numerical shape from shading and occluding boundaries", Artificial Intelligence, Vol. 17, pp 141 - 184, 1981.

6 Taylor, A., "Health monitoring agenda for a CMM machine", SME Technical Paper, MS89-527, 1989.

7 Menq, C.H., Yau, H.T and Wong, C.L., "An intelligent planning environment for automated dimensional inspection using co-ordinate measuring machines", ASME Journal of Engineering for Industry, Vol. 114, No 2, pp 222 - 230, 1992.

8 BS 7172 1989, Assessment of position, size and departure from nominal form of geometric features, British Standards Institution, U.K., 1989.

9 Walder, I. and Wallis, A.F., "Applications of 3-D solid modelling to co-ordinate measuring inspection", International Journal of Machine Tools and Manufacturing, Vol. 32, No. 1/2, pp 195 - 201, 1992.

10. Chen, Y.D., Tang, X.J. Ni, J. and Wu, S.M., "Automatic digitization of free-form curve by co-ordinate measuring machines", International Journal of Production Research, Vol. 32, No. 11, pp 2603 - 2612, 1994.

11 Ip, R W.L. and Loftus, M., "Adaptive and compensation methods in free-form surface assessment, Proceedings of the IMechE, Part B, Journal of Engineering Manufacture, 1997 (In Press).

12. Piegl, L. and Tiller, W., "Algorithm for approximate NURBS skinning", Computer-Aided Design, Vol. 28, No. 9, pp 699 - 706, 1996.

13 Watt, A., 3D Computer Graphics, Addison-Wesley, Wokingham, 1993.

14. Piegl, L. "On NURBS: A survey", IEEE Computer Graphics & Applications, pp 55 - 71, 1991.

REVERSE ENGINEERING : FROM DISCRETE POINT DATA TO 3D CAD MODEL

N.K. Tsang, K.W. Chan and S.T. Tan
Department of Mechanical Engineering
The University of Hong Kong
Hong Kong

ABSTRACT

Many reverse engineering based investigations have been reported on using the digitized surface point data of a physical object to create a Computer-Aided-Design (CAD) surface model for supporting applications such as NC cutter path generation. However, automatic creation of a complete 3D Boundary-Representation (B-Rep) model based on surface point set is still an outstanding issue since the digitized surface point data does not contain topological information.

This paper presents a method that uses the digitized surface point data of an object to automatically determine a set of planar and spherical halfspaces for representing the object. The surface point data is processed based on a shortest row gap distance criterion to form a tesselation of triangular meshes. A match-and-extend algorithm is then used to classify and merge the triangular meshes into either planar or spherical halfspaces. The bounding edges of the object can then be found by determining the surface-surface intersection of the halfspaces. A complete 3D B-Rep model of the object can thus be established.

KEYWORDS

Reverse Engineering, Quadric Surface, Surface Fitting

1. INTRODUCTION

Reverse Engineering refers to the generation of a CAD model from a physical model. In many cases, only the physical model of an object is available[1,2,3,4], examples of situation include hand-made prototype, craftworks, reproduction of old engineering objects, and sculptured bodies found in medical and dental applications. In order to facilitate CAD/Computer-Aided-Manufacturing (CAM) operations of these artifacts, it is essential to establish their CAD models.

The surface data of an object can be collected by means of direct measurement or digitizing through the use of a Coordinate Measuring Machine (CMM). Such discrete point data can only support limited applications such as copy-milling. Hence, there is a need to process the data to form a CAD model for driving various Computer-Aided-Engineering (CAE) and CAM applications such as mass properties calculation and feature information.

The CAD models formed in most current research works consist of either 1) quadric surfaces including planar surfaces and 2) parametric surfaces like Non-Uniform Rational B-Spline (NURBS) surface. It has been shown that approximately 85% of mechanical engineering components can be represented or approximated by quadric surfaces[5]. This paper is also focused on prismatic objects composed of quadric surfaces. In the present study, only planar and spherical surfaces are considered. Since the captured point data may constitute to a number of surfaces, segmentation techniques[4,6] are required to partition the data points into regions so that each region corresponds to one single surface. This paper uses a match-and-extend algorithm to automatically partition the data. Least-square fitting techniques are applied to the data in each region to generate the corresponding planar and spherical halfspaces information. By evaluating the boundary edges of these surfaces via surface/surface intersection calculations, a collection of boundary faces can be obtained. These faces can be linked together to form a B-Rep model of the object.

2. PREVIOUS WORK

Sarkar and Menq[4] used the Laplacian Gaussian operator for detecting any sharp change in shape. Trucco and Fisher[6] partitioned data in regions by using estimates of the sign of the mean and Gaussian curvature. As these methods are based on detecting sharp boundary edges and curvature changes, they are not able to detect smooth changes such as, fillet. In our work, the data is partitioned automatically by a match-and-extend algorithm so that the detection of boundary edges or curvature change is not required.

Techniques for surface fitting is an important part in CAD model creation based on reverse engineering approach. Many reported works have been based on fitting NURBS surface, but they are limited to the study of fitting a single isolated surface[4]. For fitting quadric surfaces, least-square fitting is the most commonly used method. Chivate and Jablokow[7] used a least-square based method with different eigenvalues to fit general quadric surfaces. The method can also be used to fit a specific type of surface such as a cylindrical or spherical surface. Bradley *et al.*[3] used vector equations, emphasizing the noise component, for fitting planar, cylindrical, spherical and conical surfaces.

For merging fitted faces together to form a CAD model, Chivate and Jablokow[7] determined the boundary edges by the intersection of faces, while the vertices were found by the intersection of edges under the assumption of perfect topology. However, their approach assumed that the topology of the object was already known. This assumption is not valid in a typical reverse engineering process. In our approach, the neighbourhood relationship of the fitted faces can be determined since the neighbourhood relationship of the scanned data points was established during the triangular mesh forming process.

3. METHODOLOGY

3.1 Overview

In our work, the surface point data is collected by using the automatic continuous scanning operation of a CMM. The surface of an object is scanned row-by-row (Figs. 1a & b).

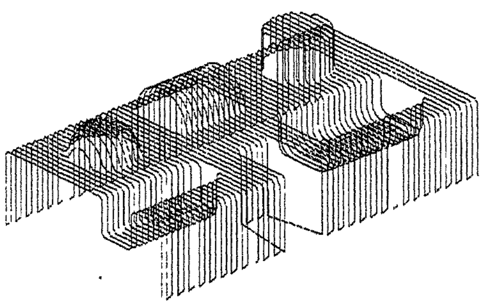


Figure 1a : The scanning path.

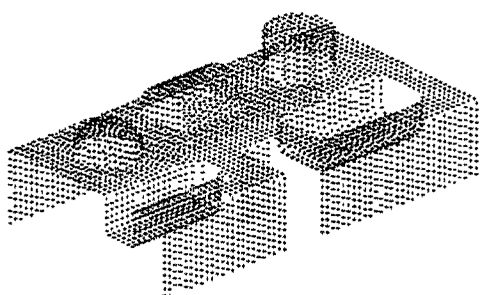


Figure 1b : Data points scanned.

The point data is processed to form triangular patches based on the criterion of shortest row gap distance which is described in section 3.2. Figure 2 shows the whole triangular grid formed. The formed patches are stored together with their neighbourhood information, i.e. their connectivity relationship. Figure 3 illustrates the data structure of a single triangular patch.

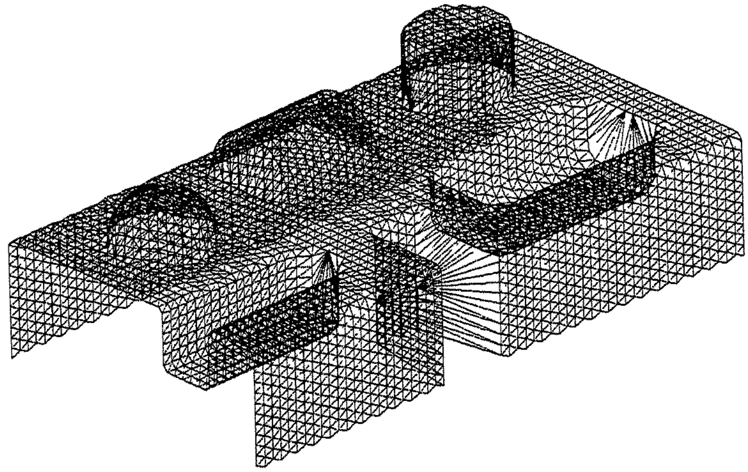


Figure 2 : Forming triangular patches.

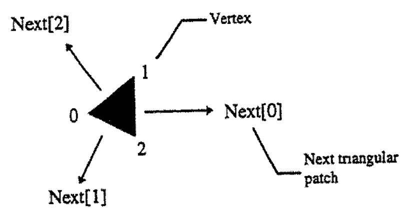


Figure 3 : Data structure of a triangular patch.

After all the triangular patches have been formed, a match-and-extend mechanism is used to extract quadric surfaces from the triangular patches. The extraction sequence adopted is 1) planar, 2) spherical and 3) cylindrical surface. The match-and-extend mechanism is described in section 3.4. At the current stage, only planar and spherical surface extraction has been implemented. The results are illustrated in Figs. 4a & b.

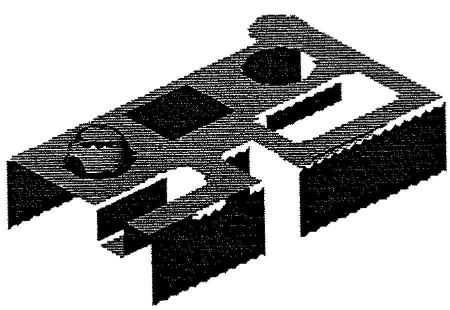


Figure 4a : Extracted surfaces.

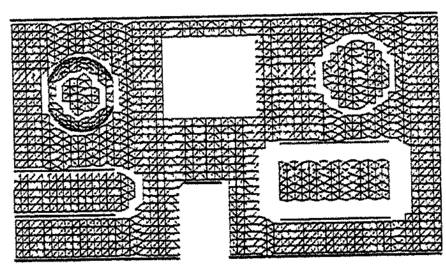


Figure 4b : A top view.

In the extraction process, the neighbourhood information among the quadric surfaces are inherited from the triangular patches, i.e. the neighbourhood relationship is preserved. When all the planar, spherical and cylindrical surfaces are extracted, the halfspaces information can be passed to a 3D CAD modelling system to perform surface-surface intersections to determine the boundary edges, thereby producing a 3D B-Rep CAD model of the object in concern.

3.2 Triangular grid formation

In practice, the number of points captured by the CMM in each row are not equal, it is therefore not possible to use the scanned data points directly to form a rectangular grid. A triangular grid is formed instead. For forming triangular patches between two rows, the data points in Fig. 5a are considered as an example. First of all, the first and the last points of the first row are linked with the corresponding points on the second row (Fig. 5b). The shortest row gap distance is then determined from the possible combinations of data points. The distance between points (1,3) and (2,4) is computed to be the shortest, hence they are linked together as shown in Fig. 5c. The remaining combinations are then divided into two portions, A and B. In each portion, the shortest row gap distance between two data points is determined (Fig. 5d). This "divide and conquer" process repeats until all the points are linked (Figs. 5e & f). Using this triangular patch generation criterion can produce a finer grid of triangular patches which better describe the surface geometry between the two rows of data points in concern.

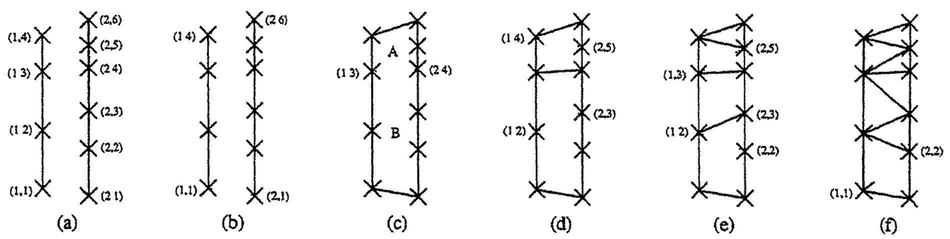


Figure 5 : Forming triangular patches based on the criterion of shortest adjacent row distance.

3.3 Mathematics used in surface fitting

The general equation for a planar surface is :

$$f_1 = ax + by + cz - d = 0 \tag{1}$$

where (a, b, c) determine the unit normal vector of the surface and $|d|$ is the distance of the planar surface from origin. This equation is linear with respect to the unknowns, linear least-square fitting based on QR-Decomposition can thus be applied to the data. The values of the coefficients in the surface equation are computed by using library routines "LAPACK"[8], which is a library of Fortran 77 subroutines for solving common problems in numerical linear algebra. The result of the equation should be divided by $\sqrt{(a^2+b^2+c^2)}$ to obtained a normalized value.

The following equation is used to represent a spherical surface :

$$f_2 = (x-c_x)^2 + (y-c_y)^2 + (z-c_z)^2 - r^2 = 0 \tag{2}$$

where $\bar{c} = (c_x, c_y, c_z)$ determine the centre of the spherical surface and r is the radius, which must be positive.

A general cylindrical surface can be represented by the following vector equation :

$$f_3 = \left|\overrightarrow{cp} - (\overrightarrow{cp}\cdot\hat{n})\hat{n}\right|^2 - r^2 = \left|\vec{R}\right|^2 - r^2 = 0 \tag{3}$$

The symbols used in the equation are illustrated in Fig. 6. The equation is non-linear with respect to the unknowns, non-linear fitting algorithms can be used to solve the problem and this is currently under investigation. The term $|\bar{R}|^2$ can be expanded and simplified into the following algebraic equation :

$$|\bar{R}|^2 = \begin{array}{l}(x-x_o)^2(1-a^2)+(y-y_o)^2(1-b^2)+(z-z_o)^2(1-c^2) \\ -2[(x-x_o)(y-y_o)(ab)+(y-y_o)(z-z_o)(bc)+(x-x_o)(z-z_o)(ac)]\end{array} \quad (4)$$

When performing non-linear least-square fitting to the data, the result of the equation should be divided by $(a^2+b^2+c^2)$ to obtain a normalized value.

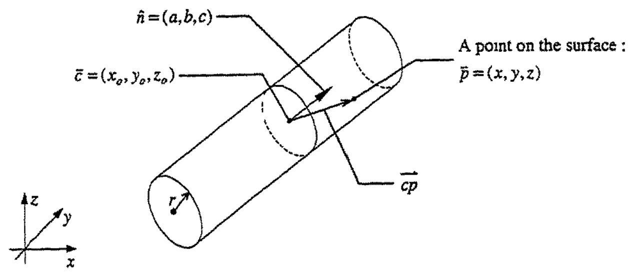


Figure 6 : Representation for a cylindrical surface.

3.4 Surfaces extraction based on the match-and-extend mechanism

The following notations are used in describing the mechanism :

t	-	A candidate patch under consideration
t_{Ref}	-	A reference triangular patch
C_t	-	A collection of triangular patches connected together with t_{Ref} as the originating patch
S_P	-	A planar surface
S_S	-	A spherical surface
T_P	-	A set containing triangular patches which can be used as representatives for extracting planar surfaces
T_S	-	A set containing triangular patches which can be used as representatives for extracting spherical surfaces

The basic procedures for extracting planar surfaces involve :

1) Selecting a triangular patch t as a candidate patch, a least-square planar surface fitting is then performed using the vertices of t together with the vertices of its three adjacent triangular patches, i.e. a total of six vertices. If the least-square error with respect to t is smaller than a predefined error value, a planar surface is likely to be located in the region surrounding the patch. This candidate patch is then put in T_P. This step is repeatedly applied on all triangular patches.
2) Select a t_{Ref} from T_P. A C_t is defined based on t_{Ref} and its adjacent triangular patches. The vertices of the triangular patches in C_t are used to fit a planar surface S_P based on the mathematics described in section 3.3.
3) A candidate patch t is selected from the adjacent patches of C_t. If the normal vector of t is equal to that of S_P (i.e. the matching process), t will be put in C_t. As a result, the number of patches in C_t is increased and the coefficients of S_P will need to be re-calculated (i.e. the extension process). This match-and-extend process is repeated with another t. Consequently, a planar surface is effectively "extracted".
4) Loop back to step 2) using another t_{Ref} to extract another planar surface.

For the extraction of spherical surfaces, the procedures are similar to that of planar surfaces :

1) Select a triangular patch t as a candidate patch, a least-square spherical surface fitting is then performed using the vertices of t together with the vertices of its three adjacent triangular patches, i.e. a total of six vertices. If the least-square error with respect to t is smaller than a predefined error value, a spherical surface is likely to be located in the region surrounding the patch. This candidate patch is then put in T_S. This step is repeatedly applied on all *remaining* triangular patches.
2) Select a t_{Ref} from T_S. A C_t is defined based on t_{Ref} and its adjacent triangular patches. The vertices of the triangular patches in C_t are used to fit a spherical surface S_S based on the mathematics described in section 3.3.
3) A candidate patch t is selected from the adjacent patches of C_t, t and C_t will share two common vertices. If the distance between the uncommon vertex of t and the centre of S_S equals to the radius of S_S (i.e. the matching process), t will be put in C_t. As a result, the number of patches in C_t is increased and the coefficients of S_S will need to be re-calculated (i.e. the extension process). This match-and-extend process is repeated with another t. Consequently, a spherical surface is effectively extracted.
4) Loop back to step 2) using another t_{Ref} to extract another spherical surface.

3.5 Implementation

The algorithms described in this paper have been coded in C-language and included as a user-function in the Unigraphics environment running on a SGI platform.

4. DISCUSSION AND CONCLUSION

The surface extraction process for planar and spherical surfaces are shown to be successful for the test-piece shown in Fig. 2. The match-and-extend mechanism applied in the extraction process can be considered as an alternative to segmentation techniques, since those triangular patches which are not recognized as a member of any of the three kinds of surfaces may be located at the boundary of two or more neighbouring surfaces.

By observing the missing planar surfaces in Figs. 4a & b, it can be seen that the CMM scanning direction affects the extraction process. For extracting surfaces which are placed in various orientation, it is desirable to scan and combine at least two scanning directions. However, the criteria for forming triangular patches will have to be revised.

For the process of forming connected faces, certain concept in feature recognition may be applied, e.g. connectivity patterns for various features, which can assist the CAD model forming process.

In merging boundary edges of connected faces, the assumption of perfect topology is essential for determining a unique vertex. Otherwise, unpredictable object may be formed.

This paper only reports the method of forming triangular patches and the extraction of planar and spherical surfaces. Further work will be done on the extraction of cylindrical and conical surfaces, and methods of forming the complete CAD model.

5. ACKNOWLEDGEMENTS

The authors would like to thank the Department of Mechanical Engineering, The University of Hong Kong for the support of the required computing facilities.

6. REFERENCES

1. Puntambekar, N.V., Jablokow, A.G. and Sommer III, H.J., "Unified review of 3D model generation for reverse engineering", Computer Integrated Manufacturing Systems, Vol.7, No.4, pp.259-268, 1994.
2. Kruth, J.P. and Ma, W., "CAD/CMM integration for reverse engineering", Proceedings of LANE'94, Vol.II, pp.705-714, Erlangen, 1994.
3. Bradley, C., Vickers, G.W. and Milroy, M., "Reverse engineering of quadric surfaces employing three-dimensional laser scanning", Proc. Instn. Mech. Engrs., Vol.208, pp.21-28, 1994.
4. Sarkar, B. and Menq, C-H., "Smooth-surface approximation and reverse engineering", Computer-Aided Design, Vol.23, No.9, pp.623-628, 1991.
5. Hakala, D.G., Hillyard, R.C., Malraison, P.F. and Nource, B.E., "Natural quadrics in mechanical design", SIGGRAPH/81 : Seminar on Solid modelling, pp.363-378, Dallas, Texas, 1981.
6. Trucco, E. and Fisher, R.B., "Experiments in Curvature-Based Segmentation of Range Data", IEEE Trans. On Pattern Analysis and Machine Intelligence, Vol.17, No.2, pp.177-182, 1995.
7. Chivate, P.N. and Jablokow, A.G., "Solid-model generation from measured point data", Computer-Aided Design, Vol.25, No.9, pp.587-600, 1993.
8. Anderson, E. ... [et al.], LAPACK Users' Guide, Ch. 1-2, SIAM, Philadelphia, 1992.

CONSTRUCTION OF SCULPTURED SURFACES FROM CMM MEASUREMENT DATA

F. L. Chen and Y. M. Chiang
Department of Industrial Engineering
Natural Tsing Hua University
Hsinchu, Taiwan, R.O.C.

ABSTRUCT

Reverse engineering (RE) approach is widely applied in design and manufacturing of sculptured surfaces. The first step of RE process is collecting data lying on surfaces. When using a coordinate measuring machine (CMM) with a touch trigger probe for digitizing data points, probe radius must be compensated according to normal vectors. Previous works were addressed on iterative measurement procedure to reduce the error caused by the compensation. Major problem of iterative methods is time consuming in digitization. This paper proposes a new architecture based on look-up table that keeps the estimated normal vectors of the measurement data to refine the data points digitized by CMM. The digitized data are first fitted into several NURBS curves by interpolation. The skinning method is then applied to interpolate these curves. Thus a smooth NURBS surface can be constructed and more accurate normal vectors of the measurement points can be obtained to compensate for the probe radius. By using this technique, we need only one digitization procedure. Therefore much time can be saved and the efficiency of reverse engineering can be improved.

KEYWORDS

Reverse engineering, Coordinate measuring machine, Probe radius compensation, Curve and surface fitting.

1. INTRODUCTION

In a typical automated manufacturing environment, the part model is usually created via computer-aided design (CAD) techniques. NC-codes for machining this part can then be generated from the design data stored in the computer. However, there are many manufactured parts with complex surfaces starting as a designer's model in clay or wood material. Typical examples are the construction of car and aircraft bodies, airplane wings, telephone sets, turbine blades, and other appliances such as tea pots, fans and toys. The design data of these objects are originally not available. In this situation, the concept of reverse engineering provides a feasible solution. The process of RE starts with the digitization from the surface of an existing part. CAD model of this part is obtained after smooth-fitting of these digitized data points. On the other hand, the design model usually needs to be modified at shop floor due to manufacturing limitations or to obtain optimal production performance. These modifications are seldom reflected in the CAD model. Therefore, RE can also be used to update new geometric models for next evaluation in the development environment.

Various measuring devices are available for obtaining the three-dimensional coordinates of a part. The digitization techniques of such devices fall into two categories: contact probing and non-contact sensing. Contact digitizers usually provide much higher accuracy and resolution than non-contact sensors. Typical examples of the contact digitizing devices are coordinate measuring machines. By touching the part surface with a probe mounted on a CMM, the x, y, z coordinates are obtained. To obtain accurate measurement results, probe radius must be compensated in the surface normal of the measurement point. CMMs typically collect the three dimensional coordinates at the center of the probe and utilize the probing directions for the probe radius compensation. If the probing direction is parallel to the surface normal, the correct measurement point can be obtained by the following equation:

$$Q = S + r \cdot \bar{P} \quad (1)$$

where Q is a point on the surface, S is the center of the probe with a radius r, and $\bar{P}$ is the probing direction.

However, when the measured target is a sculptured surface, the inaccurate compensation for the probe may cause errors to the generated model owing to the unknown normal directions of the surface (Figure 1). One solution is to use an adaptive sampling strategy to guarantee that the digitized surface accurately represents the true surface [1,2]. Since the procedure is based on the iterative measurement and refinement, the processing time to obtain the accurate part model is relative long.

This paper focuses on the RE process that begins with the CMM digitization. To solve the problem of probe radius compensation and speed up the processing time, a new iterative approach is reported in this paper. The systematical approach incorporating a curve interpolating and a surface skinning algorithm is proposed for generating NonUniform B-Spline (NURBS) surface models from three-dimensional measured data of the object. The iterative process is based on a look-up table that preserves the estimated normal vectors of the measurement data point digitized by CMM for future refinement of the surface model. Representing all surfaces as NURBS has become a tendency in CAD modeling because of the advantages of better curvature control, keeping away from round-off errors of the rational coefficients, and being a superset of many other types of surfaces.

2. LITERATURE SURVEY

Following the part surface digitization, surface fitting techniques can be applied to generate a CAD model. Two types of surface fitting are adopted: boundary interpolation and approximation. In interpolation, the fitted surface passes through all the measured data points. Boundary interpolating surface models such as ruled surface, skinned surface, and Coons surface, are widely applied in the aircraft or shipbuilding industries. In the ruled or skinned surface representation, a complex composite surface is constructed from a set of cross-sectional curves defined in one of the surface's parametric directions. The curves are usually far away from each other, therefore, a spine curve has to be provided to guide the generation of the skinned surface [3-5]. Some examples of the skinned surface representation in RE can be found in Yau's [1] and Lai's [6] works. On the other hand, approximation for curve and surface need not satisfy the given data precisely. In some applications a large number of points which contain measurement or computational noise are generated. In this case, capturing the shape of the data is important than wiggling its way through every point, thus approximation method is recommended. There are many research works addressing the least squares method to approximate a best form of parametric surface to a set of data points obtained from a digitizer [1,7-9]. Although many heuristic techniques for surface reconstruction have been reported, few papers discussed the probe radius compensation problem when using CMMs as digitizing devices.

3. REVERSE ENGINEERING PROCESSES

The idea behind our approach is using a software iterative method to obtain the accurate compensated directions rather than the iterative method proposed be Menq and Chen [2]. After fitting the compensated points into surfaces, the normal directions of each data point can be calculated. The obtained normal is more consistent with the target point's normal than the probing direction. Replace the probing directions with the calculated normal directions can result in more accurate compensated points. Refit the surface with these modified points will generate more accurate model. If the above procedure iterates again and again, the final compensated points should represent the target points as shown in figure 1 very precisely and the errors due to the probe radius compensation will be negligible. Moreover, it did not need iterative measurement thus much digitizing time can be saved.

The flowchart of our reverse engineering system is shown in Figure 2. The 3D coordinates of the part surface is obtained by contact CMM digitization. Before the part can be correctly digitized by the CMM, we must perform measurement setup such as probe calibration and setting up part coordinate system. Following the measurement setup tasks, the digitization task then starts. To match the application of the skinning method, the measured points should be sectionally distributed. The probing direction of each point is recorded in the look-up table. The iterative procedure including curve and surface fitting and error analysis then starts.

3.1 Curve and Surface Fitting

When the digitization process collects sectionally distributed data points, a curve fitting algorithm is first used to interpolate each array of data points to form a NURBS curve. All section curves can later be skinned to form a NURBS surface by means of a surface skinning algorithm.

Curve fitting process

The requirement for a set of curves to be skinned is that all curves must be comparable in degrees, number of spans and knot vector form. A *p*th-degree NURBS curve can be expressed as follows [10]:

$$C(u)=\frac{\sum_{i=0}^{n} N_{i,p}(u)\omega_i P_i}{\sum_{i=0}^{n} N_{i,p}(u)\omega_i}=\sum_{i=0}^{n} R_{i,p}(u)P_i \qquad 0\le u\le 1 \tag{2}$$

where the $\{P_i\}$ are the control points, the $\{\omega_i\}$ are the weights, and the $\{N_{i,p}(u)\}$ are the *p*th-degree B-spline basis functions defined on the nonperiodic knot vector *U*.

If C^r continuity is desired for a curve, then choosing degree $p=r+1$ is adequate. All weights are simply set to 1 most often. To pass a *p*th-degree NURBS curve through a given set of points, $\{Q_k\}$, the NURBS inversion procedure identifies the curve knot spans $\bar{u}_k$ with the interpolation points, and selects an appropriate knot vector *U*, leaving the $n+1$ control points $\{P_i\}$ as unknowns in the system of $n+1$ equations

$$Q_k=C(\bar{u}_k)=\sum_{i=0}^{n} P_i N_{i,p}(\bar{u}_k) \qquad k=0,\ldots,n \tag{3}$$

The choice of $\bar{u}_k$ and *U* affects the shape and parameterization of the curve. There are numerous papers published on the selection of the B-spline knot parameters. Three commonly used methods are equally spaced method, chord length method, and centripetal method [10,11]. In this paper, we adopted the centripetal method because it gives best results when the data take very sharp turns. To reflect the distribution of the knot spans $\bar{u}_k$, knots are selected by the averaging technique [10]. The proper choosing of knots leads to a system that is totally positive and banded with a semibandwidth less than *p*, that is, $N_{i,p}(\bar{u}_k)=0$ if $|i-k|\ge p$ [12]. Hence, it can be solved by Gaussian elimination without pivoting.

Surface skinning process

The surface skinning process means constructing a surface by blending a set of section curves. The (p,q)th-degree NURBS surface model can be expressed as follows [10]:

$$S(u,v)=\frac{\sum_{i=0}^{n}\sum_{j=0}^{m} N_{i,p}(u)N_{j,q}(v)\omega_{i,j}P_{ij}}{\sum_{i=0}^{n}\sum_{j=0}^{m} N_{i,p}(u)N_{j,q}(v)\omega_{i,j}} \qquad 0\le u,v\le 1 \tag{4}$$

where the $\{P_{ij}\}$ is a bi-directional array of control points, the $\{\omega_{i,j}\}$ are the weights, and the $\{N_{i,p}(u)\}$, $\{N_{j,q}(v)\}$ are the nonrational B-spline basis functions defined on the knot vectors *U* and *V*.

In this paper the skinned surfaces are constructed using the method of interpolation. The (*p*, *q*)th-degree NURBS surface with all weights equal to 1 is good to represent the model. If one of the parametric variables, say v, is fixed, i.e., $v = \bar{v}_l$, $S(u,v)$ can be reduced to an isoparametric curve in terms of u. When $v \in [0,1]$ is divided into l segments, the surface can be divided into l isoparametric curves which can be expressed as follows:

$$C^l(u) = S(u,\bar{v}_l) = \sum_{i=0}^{n} N_{i,p}(u) R_i(\bar{v}_l) \tag{5}$$

where $R_i(\bar{v}_l) = \sum_{j=0}^{m} N_{j,q}(\bar{v}_l) P_{i,j}$ (6)

This indicates that $R_i(\bar{v}_l)$'s can be considered as the control points of the isoparametric curve $C^l(u)$. The surface skinning algorithm used in this paper is doing m+1 curve interpolations first t to obtain the section curve $R_{i,l}$, then doing n+1 curve interpolations to obtain the control points $P_{i,j}$.

3.2 Error analysis

Suppose any point lying on the surface with the parameter (u, v). In the digitization task, we kept the probing direction in the look-up table. The probing direction is viewed as the normal vector and used for probe radius compensation. Denote the vector as $\vec{N}'_{uv}$, and the probe center position as $S(u,v)$. The relation between measurement point $Q(u,v)$ and $S(u,v)$ is

$$Q(u,v) = S(u,v) + r \cdot \vec{N}'_{uv} \tag{7}$$

where r is the probe radius

Once we obtain the surface normal vectors $\vec{N}_{uv}$ from the fitted model, we can use it to refine the measurement point by the following equation:

$$Q'(u,v) = S(u,v) - r\ \vec{N}_{uv} = Q(u,v) - r \cdot (\vec{N}_{uv} + \vec{N}'_{uv}) \tag{8}$$

If the probe compensation process causes large error, the deviation between $Q(u,v)$ and $Q'(u,v)$ will be large. Let $\overline{D}$ be the average deviation between $Q(u,v)$ and $Q'(u,v)$, i.e.,

$$\overline{D} = \frac{\sum_{i=0}^{n}\sum_{j=0}^{m}(Q(u_i,v_j) - Q'(u_i,v_j))}{(n+1)\times(m+1)} \tag{9}$$

If $\overline{D}$ is large than a specified value ε, then we replace the measurement points $Q(u,v)$ with $Q'(u,v)$ and refit the surface again. At the same time of the iteration excusing, the probing directions kept in the look-up table are replaced by the calculated surface normal vectors. The process will repeat until the deviation converges to a predetermined value.

4. EXPERIMENTS

In order to verify the effectiveness and efficiency of the iterative method developed in this research, several examples were experimented in the laboratory. Mitutoyo CHN303-06 coordinate measuring machine with a PH9 probe is served for the digitization task. The CMM has an accuracy of 1 *um* and a repeatability of 1 *um*. The stylus of length 20mm and diameter 2mm is mounted on the probe. Before digitization, the probe tip was calibrated by the master ball with diameter 1 inch.

A clay model of a complicated sculptured surface with a size 150 *mm* x 150 *mm* was experimented as an illustration. Before digitization, the probe tip was calibrated and obtained the tip radius of 0.9975mm. The part surface was divided into 15 cross-sections in Y direction and 15 points were digitized in each cross-section. Figure 3 shows the measurement result of the complicated surface after probe radius compensation. Points on each cross-section are first interpolated to a NURBS curve with degree 3, then skinned to a NURBS surface with (u, v) degree (3,3). In this example, we specified the stopping deviation criteria as 0.001mm. After three iterations, the average deviation is reduced below the specified value. The final compensated points are used to fit the surface model. Table 1 shows the errors of each iteration. The final fitted surface model is shown in Figure 4.

To verify the accuracy of the fitted surface model, we retrieve all interpolating points and their normal vectors to generate a CMM measuring program file. The CMM program probes each data point in the direction of surface normal thus the compensation will be correct. After executing the program, the result was compared to the surface model and the first digitization, and obtained the corresponding average deviations of 0.00404 *mm* and 0.40864 *mm*. The error between the measurement data and the fitted model is quite small. It means that the fitted surface adequately represents the model. Besides, when compared the measurement data with the first digitization, the average deviation is very close to the value of the first iteration (see Table 1). This indicates the improper probing direction causes the error radius compensation and it can be solved by the proposed method without iterative measurement.

5. CONCLUSION

A surface skinning method, incorporating a software iterative procedure, was proposed in the work for the reverse engineering of sculptured surfaces. The improper probing directions during the CMM digitization generally induce significant errors caused by the probe radius compensation process. To solve the improper radius compensation problem and decrease the processing time of getting correct model, this research developed an iterative method based on look-up table which keeps the estimated normal vectors of the measurement data to refine the data points digitized by CMM. The digitization process is performed on several cross-sections of the surface. The measurement data is first fitted to several NURBS curves by interpolation, then skinned to a NURBS surface. Experiments indicate the proposed method can correctly represent the surface model and reduce much iterative measurement time since the digitization task is performed only once. Future research will focus on development of better digitizing process for accurate and efficient data recording, and studying for reconstruction of complex objects with discontinuous features.

6. REFERENCES

1. Yau, H. T., Haque, S. U. and Menq, C. H., "Reverse Engineering In The Design of Engine Intake and Exaust Ports", Manufacturing Science and Engineering, Vol. 64, pp. 139-148, 1993.
2. Menq, C. and Chen, F. L., "Construction of CAD Models from CMM Measurement Data", Proceedings: 3rd International Conference on Die & Moiuld Technology, pp. 771-780, Taiwan Mold & Die Industry Association, Taipei, Taiwan, 1995.
3. Tiller, W., "Rational B-splines for Curve and Surface Representation", IEEE Computer Graphics and Application, Vol. 3, No. 10, pp. 61-69, 1983.
4. Filip, D. J. and Ball, T. W., "Procedurally Representing Lofted Surfaces", IEEE Computer Graphics & Application, Dec., pp. 27-33, 1989.
5. Woodward, C D, "Skinning techniques for interactive B-spline surface interpolation", Computer-Aided Design, Vol. 20, No. 8, October. pp. 441-451, 1988.
6. Lai, J. Y., Lin, C. Y., and Liou, C. S., "A reverse engineering approach for complex shape modeling", Proceedings: The Fourth International Conference on Automation Technology, Vol. 1, pp. 231-238, ITRI, Hsinchu, Taiwan, R.O.C., 1996.
7. Duffie, N. A., and Feng, S. C., "Modification Bicubic Surface Patches using Least Squares Fitting Techniques", Computers in Mechanical Engineering, CIME Research Supplement, September, pp. 57-65, 1985.
8. Hoschek, J, "Intrinsic Parametrization for Approximation", Computer Aided Geometric Design, Vol. 5, pp. 27-31, 1988.
9. Sarkar, B, and Menq, C. H., "Smooth Surface Approximation and Reverse Engineering", Computer-Aided Design, Vol. 23, No. 9, pp. 623-628, 1991.

10. Piegl, L., "On NURBS: A Survey", IEEE Computer Graph and Application, Vol. 10, No., 1, pp. 55-71, 1991.
11. Lee, E. T. Y., "Choosing nodes in parametric curve interpolation, CAD, Vol. 21, pp.363-370, 1989.
12. De Boor, C., "A Practical Guide to Splines", Springer-Verlag, New York, 1978..

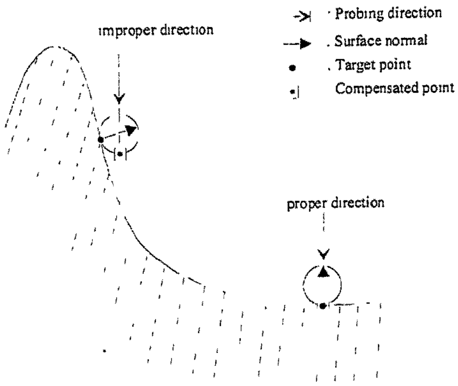


Figure 1. Probe radius compensation

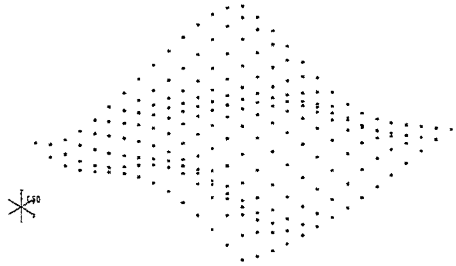


Figure 3. Measurement data points of the sculptured surface

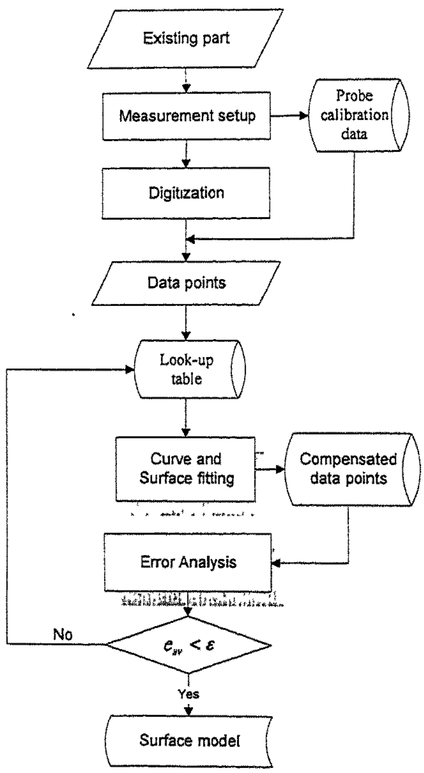


Figure 2. Flowchart of the RE system

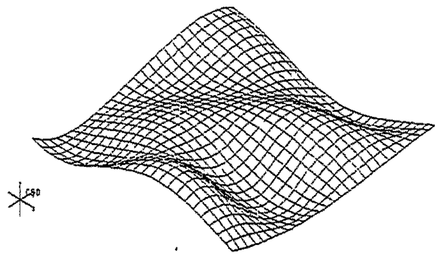


Figure 4. Fitted surface model

Table 1 The error of each iteration of the sculptured surface

# of iteration	average deviation (*mm*)	max. Deviation (*mm*)
1	0.41043	0.93240
2	0.00185	0.02870
3	0.00010	0.00138

An Assistant Model in C.A.D. Systems Based on Interactive Planning

Lotfi Bellalem, Yvon Gardan & Abdelouahed Zakari
Laboratoire en Informatique de Metz
Ile du Saulcy
57 000 Metz - France
{bellalem, gardan, zakari}@ lrim.univ-metz.fr

ABTRACT
The introduction of Artificial Intelligence techniques has contributed to the evolution of CAD/CAM systems. The domain of mechanical engineering, in general, and methods of assembling, in particular, profit by this technique.
However, proposed systems often use system expert approach which aims to provide general techniques of assembling where the user remains passive, or when it is not the case, his participation is reduced to modifying the parameters of the assembling and do not concerns
the process of assembling itself.
In this paper, we propose an assistant model aiming at providing sequences of assembling in interactive way. This model uses mechanisms of interactive planning which is a tool to manage the dialogue "system/user" and the design process. The user operates as a partner and not as passive actor. The choice of this approach is an alternative of knowledge based systems dedicated to design in CAD systems.

KEYWORDS
Planning, Dialogue, CAD, Knowledge Base, Artificial Intelligence.

1. INTRODUCTION

The objective of this work is to demonstrate that a tool as interactive planning can contribute to increase the interactivity in the CAD systems dedicated to mechanical engineering either at the level of assembling or at the level of objects building.
With this aim, a consequent environment is associated to this tool which includes the following functionalities:
1- a multi-level and modular architecture,
2- a design process adapted to the user type (expert or not),
3- a dialogue manager,
4- adaptability to the knowledge base.
The defined system is associated to a modeler such as B-Rep (Boundary Representation) or CSG (Constructive Solid Geometry), for instance. The modeler characteristics are only considered when the system produces a sequence of commands in the form of predicates which are adapted depending on their semantics to the operative type of the modeler.
Before presenting this model, a brief description of the problematic of the association between the IA and CAD is proposed.

2. PROBLEMATIC OF THE ASSOCIATION BETWEEN IA AND CAD

Systems based on Artificial intelligence techniques in CAD appeared because, on one hand the technical progresses in the CAD domain incite to propose more elaborated techniques and on the other hand, because the time given to manpower is more and more reduced.
This has as consequence to find mechanisms to reduce the user's charge in this activity of designing where analysing and synthesise are associated.

2.1 Expert systems

The designing systems based on knowledge bases are well adapted in the framework of expert applications including an important part of know-how and where the designing process is relatively stable and well controlled (*cf.* SMECI [5]). These systems also allow the automation of repetitive phases of designing, the user takes the charge of more creative tasks and the final choices of designing.

The current tendency of expert systems is SE2G (expert systems of second generation) where different models of domain and different methods of resolution are necessary depending on the type of the considered problem. Two approaches can be distinguished in this type of systems. The first emphasises the collaboration between several kinds of knowledge and it is designated by heterogeneous systems. The second one uses several methods of resolution and it is designated by the knowledge level.
We can mention as heterogeneous systems ABEL [6] and HEARSAY II [7]. More recent works highlighted the possibility to resolve a problem using another type of knowledge which correspond to the notion of model domain and the associated resolution methods. For instance, GTD system (Generate, Test and Debug) [8] has been developed by R. Simmons at MIT in order to resolve problems of interpretation or planning of the shape i.e. to built a sequence of events which leads from an initial state to a final state.

Concerning the knowledge level, we can mention KADS (Knowledge Acquisition and Design Structuring) which is a methodology of designing of knowledge based systems which has been developed from certain works of Amsterdam University [12], COMMET which is the result of L. Steels [9, 10, 11] Works at University of Bruxelles in the framework of the project ESPRIT.
COMMET allows to describe an application according to three aspects: the tasks, the domain model and the methods. Several models organise and structure these entities.

However, it remains to provide to these systems the appropriate interface ie to integrate human-computer dialogue mechanisms which must be as good as the conceptual models which are appearing with SE2G. In this objective, two problems must be resolved:
–to provide intelligent functionalities i.e. inference mechanisms which can guide a non specialist user;
–to take into account the designer intentions and to propose him alternatives of designing at each time if it is necessary.
To resolve these two difficulties, it seems to us that the proposed system must possesses the following properties:

1. to propose different ways during the designing process depending on the user type (expert, or not) and do not impose a precise alternative without a rigorous negotiation.
2. to offer an environment which manage the different information which are produced during the designing process.
3. to bring reasoning mechanisms called intelligent into operation (inference systems).
4. to offer learning mechanisms so as to allow on one hand the evolution of the system, and on the other hand the initiation of the non expert user relatively to the difficulties of the designing.

2.2 Planning

The most usual domain which uses planning is the planning of automated tasks that the most representative is the robotics. The principle of planning is known for a long time and it is modelled in a space of problem containing the following elements: a initial situation, a set of final situations, a set of operators allowing to pass from a situation to another.
On the other side of the mechanisms which improve a planning system making it « intelligent » (because of inferences mechanisms , heuristics knowledge acquisition, etc.), it is necessary to think about the problem of probability. As a matter of fact,, most of planning systems as expert systems) carries the properties of the domain for which they were built and every transfer to other domain involve a diminution of performance. For a CAD application, and in particular for the engineering

mechanisms, the idea is to preserve at the root a classical planning (i.e. finding an operator in order to pass from an initial state to an another state with inference mechanisms) and next to add interactive mechanisms which are specific to the treated domain with information concerning the user type (expert, no expert). This is what we call *planning dialogue* which depends on the user choices. In order to complete this dialogue system, an environment containing a knowledge base and a functional base must be added as it is presented below.

2.3 Assembling in mechanical engineering

In this domain, every product appears as the individual union of indivisible components and the result produces the product itself. Such as in robotics, automated assembling of an object implies essentially two activities: Generation of the appropriate assembling sequence and programmation of the robot.
Generation of assembling sequence consist in the determination of the precise assembling order using different combination of the object components. This order often depends on geometrical characteristics of the element. As a matter of fact, the geometry of the element contributes to reduce significantly the assembling order.

Works in this research domain are important but they often use the same principle. For instance, Bourjault and Whitney [1] [4] propose to model assembling using a graph of liaison which is, by definition, a simple graph where a line correspond to functional contacts between the components to join. The user gives a list of preceding relations which constraint the order of liaison in the construction process of the product. These preceding relations are defined by the answers to a set of questions proposed by the system which produce all the assembling mechanical sequences relatively to the preceding relations given by the user.
Other studies such as De Mello and Sanderson studies [2] are based on a decomposition approach that means that the product is transformed in disassembling sequences series. This approach begins with building up the relational model of assembling in a graph where lines are in contact with surface contact or fixation and where arcs represent the different relations between couple of lines. Another approach is proposed by Laperriere and ElMaraghy [3] where the relations between components are modelled interactively through a diagram of relation.

Our work combine the use of knowledge bases dedicated to CAD and assembling methods based on graph of liaison. This model proposes, from information concerning the user type, a set of possibilities of assembling or designing, with focalisation on the dialogue aspect. The architecture is modular, and this allows us to consider others applications in the context of mechanical engineering

3. THE PROPOSED MODEL AND ITS ARCHITECTURE

We propose a model which is the combination between interactive planning, knowledge based systems, and dialogue management (*cf.* figure1). The aims is to satisfy the user according to his experience. Essentially, the proposed model is centred on using interactive planning between the system (knowledge-based + functional-based) and users having different experiences. The set of users are classified as follow:
<u>Expert user</u>: The system provides different alternatives most numerous and most varied.
<u>Not expert user</u>: The system provides mechanisms of negotiation for helping the user to do his choice, and managing versions of design.
This system is an assistance model dedicated to CAD application and it is coupled to geometric model for visualise design condition. Actually is B-Rep, but in our architecture we can use different modeler as CSG or other modeler.

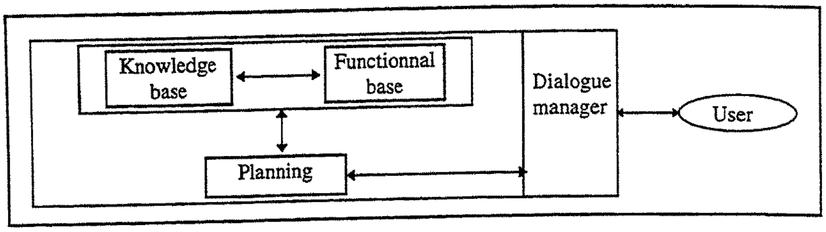


Figure 1- Global architecture system

3.1 Knowledge base

The knowledge base contains the relationships, the data and the representation of rules and a set of constraints.

A- Category of data

The knowledge base is composed by three categories: the product, the geometrical characteristic and the components.

1- The product is a composed object, it is constructed from components assembling.

2- The components are two types: classical (for example pin, screw, etc..) and not classical components.

3- The geometrical characteristics are defined from three types of base: cylinder, hole, block.

A set of parameters is associated to each family, also its characteristics. Each object is represented such as set of predicates and each list stock. Some individually information concerning an object.

B- Relationship between data and diagram of components

The relationship between the data are defined although a relationship diagram, itself constructed from two relationships:

1- Physical connections and spatial constraints. Its relationships exists between the components. The physical connection between two components imposes a contact between three components. For this reason, there are three types of physical connection: *screw in, against, adjust.*

2- The spatial constraints allow to define the relationships of previously between different components, so there are three types of spatial constraints: *in, recover, recover and in.*

the relationships of diagram is structured using the connection physical operators and spatial constraints operators. It represent internal structured of initial and final state for each components.

For example (*cf.* figure 2 and figure 3):

List of parameters: (a, adjust), (c, against), (r, recover).

List of predicates of initial state: Against(Base, table), Against(Plate, Base), Against(Pin, Base) + The rules of physical compatibility.

List of predicates of final state: Against(Plate, Base), Recover(Plate, Base), Recover(Pin, plate), Adjust(Pin, Plate), Adjust(Pin, Base).

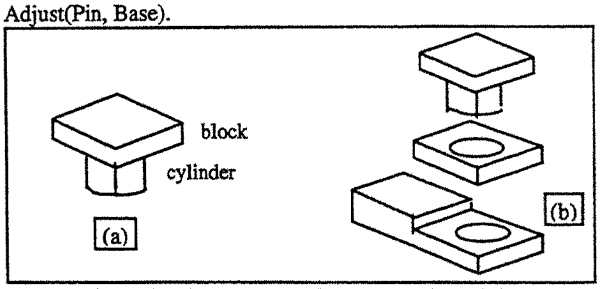


Figure 2- (a) Schema of pin - (b) Blocks to join

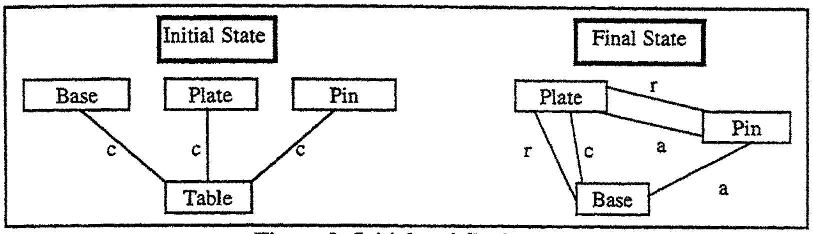


Figure 3- Initial and final state

C- The rules of physical compatibility

To check the validity of physical connections between two components of whole assembling mechanical part, the geometrical characteristics of these components are extracted and also analysed through a list. So, in our example to validate, the physical connections between the pin and the holes they must be compatible either in geometrical or dimensional way.

For this achievement, we apply the rule (for example):

if (a characteristic is a full hole)
and (the other characteristic is a full cylinder)
and (the diameters of the hole and the cylinder are compatibles)
then (Propose that the physical connection is valid)
else (Take another pair of characteristics).

3.2 Functional base

Two main functions are associated to the functional base:

- it analyses and possibly validates a product as a candidate by using specific procedures for different types of users.
- it stores different validated alternatives associated to a design assembling session and this through more or less successful versions depending on the complexity of the assembling part. The functional level (*cf.* figure 4) contribute to the system evolution and also to define more precisely the model of the assembling design chooses. The main idea is to allow the user not only to learn but also to enrich the system with new knowledge resulting from assembling session. In next figure, we describe the architecture of modules which are necessary to build mechanisms.

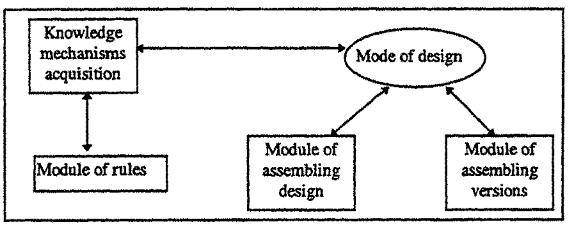


Figure 4- Functional base

3.3 The planning module

The planning module takes a mainspring functionality in the whole assisted system. Its goal is to provide one or several actions plans when user formulates a request to the system.

A- The planning Architecture

There are two parts that constitute the planning module: the plan generator (or generator for short) and the inference engine. The generator is the algorithmic part, its main function is the identification of operators witch can reach a giving objective. The generator gives the representation of minimal knowledge needed to construct the plan. The inference engine enable to reduce the research space of the solution to choose objects that it bind to variables of operator. We can add new rules to the inference engine as and when we have knowledge acquisition, so it can increase the process planning efficiency and the quality of the generated plans. The proposed architecture is based on experiments of inference engine (first order) which refine and increase this experiments, with a view to have an optimum of the whole behaviour of the system (*cf.* figure 5).

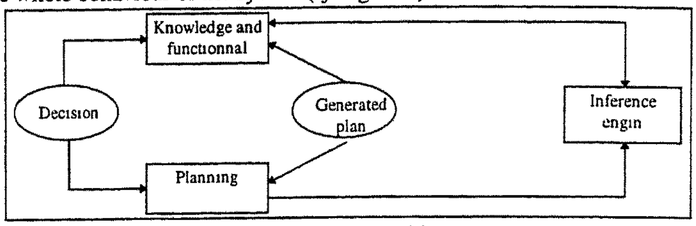


Figure 5- Planning architecture

The generator strategy is breadth first, in fact this enable to find in every case the path if there exist. In this way the generator chooses the most left node to develop. The planning module use a search tree construction, where each mode is a representation of the state of plan in one stage of its evolution.

B- The knowledge acquisition

The experiments constitute the only source of knowledge for the planning module. So, in the cycle learning/planning the learning mechanisms are carrying out subsequently. In other words, all alternatives have been checked, the planning module knows the result, and for each them: fails or achievement of a plan.

3.4 The Dialogue Manager

The dialogue manager contribute to take in account the user requirement with the system facilities by enabling him to formulate his request better and this by negotiation. The dialogue manager constitute the interface that should not be missed, between the user and the system. The three main functions of the dialogue manager are:

- to assist the user to formulate his request;
- to complete his request by adding missing information;
- to restart the procedure when the user is dissatisfied.

A - The Dialogue Procedure

We can summarise the procedure an follow:

- The user formulate his request;
- The manager ask him to precise the goal if there exist any ambiguity;
- When the generator is active, it should have, if it necessary to complete a request and if not to reject this request;
- The Manager gives a first solution to the user, and then we have two cases:

1 - The user agree to the solution, in this case the system store the representation of the track and then propose to the user to continue with the current state (of the current extension of the solution) as an initial solution.

2 - The user dose not approve the solution, in this case the system continue to spreading out the tree of actions until he gets an other solution.

B - The historical account of the dialogue

By keeping each time a track of the path (in the tree) chooses by the user, we constitute in fact, a historical account of the dialogue as pointed in figure 6.

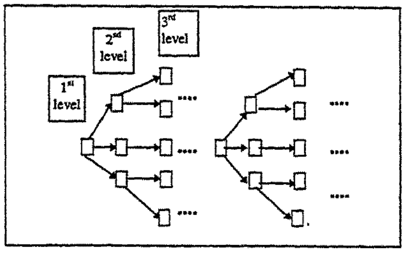


Figure 6- Representation of dialogue historic

4. CONCLUSION

In this paper, we have tried to demonstrate the necessity of a global assistance model in the domain of assembling engineering. This model combine specific mechanisms of interactive planning with those of the knowledge base and those of dialogue managing. The main objective is to satisfy the user depending on his type. For an not expert user, the aim is to propose him mechanisms of returns in important number. To realise this, two tools are needed: the negotiation through the historic of dialogue, and the managing of assembling versions. These tools allow to user to learn and thus to reuse this experience in another assembling session. For an expert user, the aim is to propose him alternatives more rich and more numerous. In this case, three tools can be considered:

a- the interactive planning tool which provide sequences more numerous if the user want it,
b- the dialogue manager tool which decrease the number of returns,
c- the knowledge base tool which provide, through the versions manager, a procedure of quick consultation, in particular for complex objects.

References

[1] Bourjault A., "Contribution à une approche méthodologique de l'assemblage automatisé: élaboration automatique des séquences opératoire", Thèse de doctorat de l'Université de Franche-comté, France, 1984.

[2] Homem De Mello L. S., Sanderson A. C., "Automatic generation of mechanical assembly sequences", Proceedings of IEEE Internationnal Conference of Robotic and automation, Scottsdale, Arizona, pp 56-61, 1989.

[3] LaperriereL., ElMaraghy H., "Automatic generation of robotic assembly sequences", International Journal of Advanced Manufacturing Technology, 1991.

[4] Whitney D. E., Defazio T.L., Gustavon R.E. , Graves S. C., Cooprider K., Holmes C. A., Klein J.C., Lui M., Pappu S., "Computer aided design of flexible assembly systems", Report n°CSDL R-2033, C.S. Draper Laboratory Inc, Cambridge, Mass, 1988.

[5] Haren P. , Neveu B., Corby O., Montalban M., "MEPAR un moteur d'inférence pour la conception en ingénieurie", 5ème Congrès de RFIA 1985,

[6] Patil R. & al., "Causal understanding, of patience illness in medical diagnosis", IJCAI 1981

[7] Erman L, et al, "The Hearsay II speech-understanding systems: integrating knowledge to resolve uncertainty"; Computing Survey 12 (12), 1980.

[8] Simmons R., & Davis R., "Generate, Test and Debug: Combining associational rules and causal models"; Proceedings Tenth IJCAI, Milan 1987.

[9] Steels L. & al, "The CONSTRUCT project: what we have done; what we would like to do"; Delivrable CONSTRUCT; 1992

[10] Steels L.,"The Componential Framework and its Role in reusability"; SGES-93. 1993.

[11] Steels L., "COMMET: A Componential Methodology forknowledge Engineering", Deliverable CONSTRUCT, 1991.

[12] Wielinga B. and al, "Towards a Unification of Knowledge Modelling Approaches"; SGES-93.

BUILDING A HOMOGENEOUS DESIGN ASSISTANT FROM HETEROGENEOUS EXPERTS: RESULTS FROM THE IMCOD PROJECT

Ansgar Bernardi, Bernd Bachmann, Anna-Maria Schoeller
German Research Center for Artificial Intelligence - DFKI GmbH
P.O. Box 2080
D-67608 Kaiserslautern, Germany
Tel. ++49-631-205-3582, Fax ++49-631-205-3210
bernardi@dfki.uni-kl.de

ABSTRACT
The IMCOD project (Intelligent Manager for COmprehensive Design) developed techniques to cope with the problem of the optimal use of existing information systems for design support. We present some central results of this research: Existing support systems are considered as so-called *local experts* which are integrated under guidance and control of a central managing system. To the user the managing system is presented as a homogeneous design assistant which is able to supply the requested information during the design process. Internally the manager selects the relevant local experts, enables the necessary communication, and provides their results to the user at the right time.

KEYWORDS
design, system integration, decision support, communication model, intelligent assistant

1. INTRODUCTION

1.1 IMCOD's Motivation by the Application

Modern industry faces some challenging requirements: Modern products must offer ever-increasing quality at market-conforming prices. Increasing numbers of different product variants and the constant request for shorter response times and the shortest possible time-to-market ask for new flexibilities in every facet of the production process. Additional constraints arise out of the political field: Growing environment–awareness leads to new laws and tends to underline a companies responsibility for the whole life–cycle of their products. Consequently problems like recycling of the product and environmental impact of the production process have to be taken into account. As most of the decisions concerning these requirements are made during product design, there exists a growing demand for effective information handling during the design phase.

Although information systems like databases, variant retrieval systems, FEM systems or CAPP systems provide valuable support by solving difficult or time-consuming tasks, they contribute to another problem which severely hinders their utility: The task to employ the growing number of various support systems in an optimal way and at the right moment is in itself a task of increasing complexity. This is aggravated by modern organizational structures: A product-centered development process controled by one responsible product manager but encompassing many different and possibly distributed experts, supporting systems, and tools is a very complex system; to employ the various information sources in an optimal way is a very demanding task.

The challenge to modern product design is thus a problem of information transfer: The person responsible for the product development (in modern organizational structures often known as *product manager*) needs all relevant information about every aspect of the product's life-cycle just at the right moment during the design phase.

1.2 The IMCOD Approach

To tackle these problems IMCOD realizes a so-called *manager* which organizes and manages the cooperation with the different design–supporting systems, called *local experts.* (Please note that the term *local expert* does not necessarily imply anything special about the nature of these systems. The spectrum may rise from conventional databases to sophisticated expert systems. Even communication interfaces to human experts involved in the design process fit into the scenario.) The IMCOD manager mediates between the human responsible for the product development and the different supporting systems (see Fig. 1).

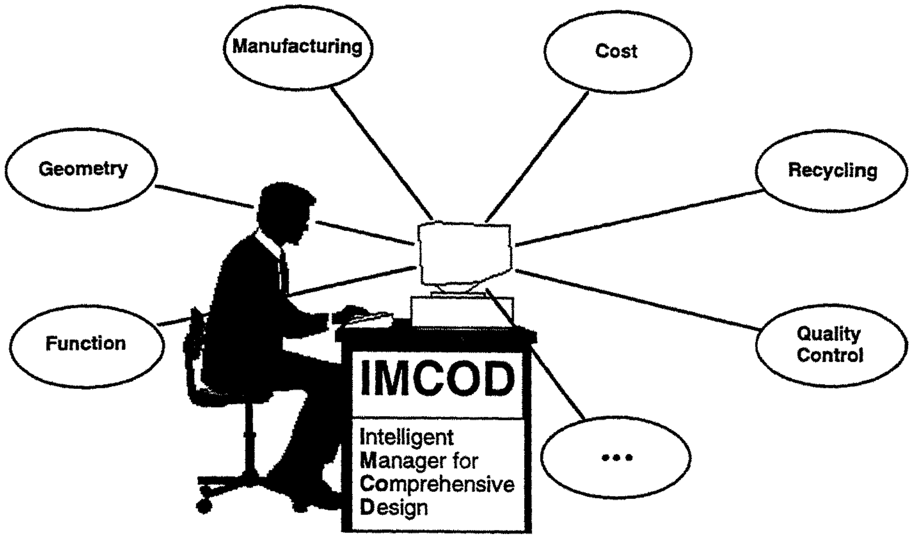


Figure 1: IMCOD mediates between human user and supporting systems

In order to realize this approach the IMCOD project contributed to three central topics:

- The manager has to select competent local experts at the right moment during the design process and must provide the necessary information to enable the local experts to perform their intended functionalities. IMCOD monitors the design process and automatically activates the relevant local experts.
- The manager has to translate between the diverging ontologies of the various experts. IMCOD provides a semantic unification approach [1] to this problem.
- The manager must integrate the various contributions of the local experts in order to provide homogeneous support to the product design. IMCOD amalgamates the contributions of the local experts by calculating a multi-criteria decision model.

In summary, IMCOD integrates different heterogeneous design–supporting *local experts* into a homogeneous design assistant.

2. SOME DETAILS ON THE IMCOD APPROACH

2.1 IMCODs Vision of an Intelligent Assistant

The goal of the IMCOD project is to integrate existing design-supporting systems into a homogeneous design assistant. An *intelligent assistant* should interact with the user closely and in a natural way, which poses special constraints on the user interface and the specific terminologies to be employed. The system must be able to globally understand the task at hand and, as a consequence of the global understanding, the system should be able to decide which task to apply at any given time. Finally the assistant should isolate the user from the technical particularities of the incorporated modules.

In summary the vision of an intelligent assistant results in a new division of labour among the human user and the supporting system. The system may perform several tasks, might decide on its own which task is needed at a given moment, and focuses on support of the global workflow. The crucial goal which must be achieved is the necessary user acceptance. The success or failure of an intelligent design assistant depends on whether the user really feels supported.

The IMCOD approach to reach this vision is the integration of several existing design-supporting systems into a homogeneous entity The specialized capabilities of the different local experts remain untouched. The manager component provides the necessary knowledge about the global process as well as the integration under a common user interface and the necessary terminological and decision-theoretical amalgamation.

2.2 Integration of Existing Information Systems

In order to enable the reuse of existing systems, IMCOD realizes a communication approach. However, there are some obstacles that hinder its realization even if the technical prerequisites for the communication are given [2]: Since there is no 'best suitable' syntax for different sorts of information, different systems use different languages of incompatible dialects within one syntactical language paradigm. Information sharing can be impossible if two systems are build on different conceptualizations, e.g. different sorts of value types. The STEP [3] debate illustrates that these difficulties arise even within a single standard due to necessary subdomains .

In order to overcome these obstacles IMCOD adapted the *software engineering model* to provide a methodology that enables translation between different models. The software engineering model regards information systems as software modules with an input–output and a functional specification which can be integrated as a 'black box' into other systems. This approach results in the communication model that we use in IMCOD [4], cf Fig. 2 lower part:

- The *coordination layer* fixes the addresses of the sender and the recipient, the identification of the messages, etc.
- The *communication layer* defines the logic of the communication, i.e. the message type, the quality of the message, etc.
- The *content layer* specifies the content of the message, both syntactically and semantically.

Each of these layers has to be specified to achieve interoperability in a scenario as sketched above. The layer that interests us most is the contents layer. It defines how the contents of the message looks like both syntactically and semantically. For the syntactic part there are some well known languages that have proven to be suitable to capture a wide range of languages and dialects, e.g. predicate logic, KIF (Knowledge Interchange Format) [5], EXPRESS, etc. In the context of this work we will use predicate logic as the interlingua. However, the existence of an interlingua is only a prerequisite for the information sharing. The reason is that though two information systems may use the same interlingua and thus do correctly interpret the structuring primitives of the language they might not interpret the identifiers accordingly.

Within the context of information sharing the notion of *ontologies* is used to establish a means for a common understanding of information among different agents. More specifically, an ontology (of a domain) may be regarded as the definition of the conceptualization that is shared among different agents: the basic set of objects that exist in a domain and the relationships among them. As an example STEP can be regarded as an ontology for some limited domains whereas EXPRESS is the according interlingua.

Taking into account the above observations we have to translate messages that are based on one conceptualization into messages that are based on a different one. We developed a solution approach that basically consists of three steps, cf Fig. 2:

- First, we provide a knowledge representation language and a system that allows for the definition of ontologies for each of the participating information systems individually.
- Secondly, we define a methodology for describing translation rules between terms of the two ontologies at hand.
- Thirdly, we apply the translation rules at run time to achieve a semantics preserving transformation of messages that are send from one information system to the other one.

To achieve this goal we build on some technologies and techniques that are well known. The representation language for the ontologies is based on the *Ontolingua* approach [6]. For the purpose of describing the translation rules we use the approach developed in mathematical logics of theory interpretation[7]. Finally, the intrinsic translation step for messages sent from one information system to the other is then a straightforward application of a sequence of structure preserving translation steps that are applied to the message to be sent to the other information system.

We sketched how our approach defines a methodology for the integration of the heterogeneous local experts based on a specific communication model. It is based on techniques that are generic enough to be applicable to a wide range of existing systems. To achieve the intended IMCOD functionality the methodology for integrating the local experts on a communication basis must be enhanced by a technique to evaluate the answers - so far just expressions in a specific language - on the background of the design task at hand.

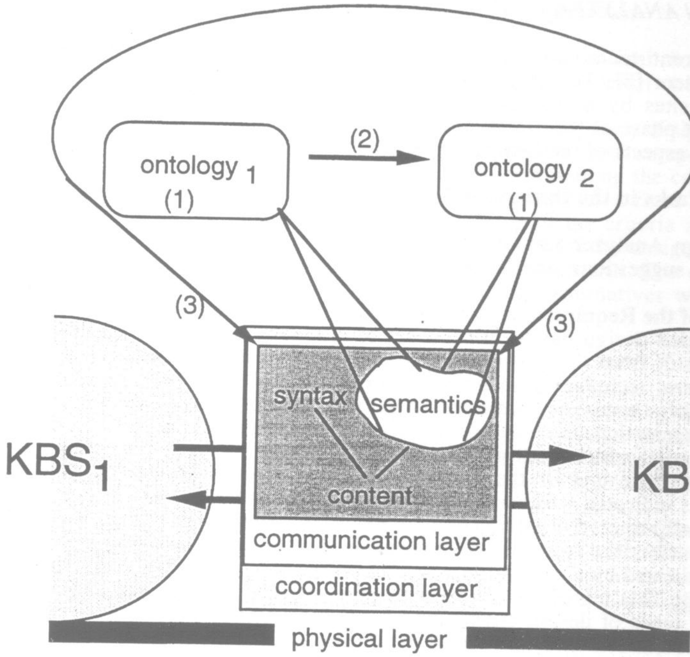


Figure 2: IMCOD's communication model and the principle solution approach

2.3 Decision Support in IMCOD

Decision support in general comprises two different approaches: First, the human decision maker may be supported by providing access to information on the relevant level of detail. The goal of a decision support system in this approach is to facilitate the access to the information sources, to filter for information necessary for the task at hand, to condense or expand information where necessary, and to mediate between different abstraction levels. Thus the system has to bridge the gap between detailed descriptions of the real world and the abstract terms of a decision makers realm.

The second possibility of decision support is to perform decisions by calculations according to well-defined criteria. Thus part of the human decision makers expertise is formalized by some mathematical methods. The explicit description of the parameters which influence the decision and the representation of the decision makers goals and preferences by utility functions allow for the automatic calculation of the decision. On this basis a decision support system is able to evaluate and rank different alternatives in accordance to the same criteria the human decision maker would apply. A system of this kind is well suited to choose among many different alternatives, taking into account many parameters, and to provide an optimal result in a sound, well-founded and documentable way.

The IMCOD project tackles both approaches of decision support. The communication management, the terminology handling and the global control of the design process provide integrated access to different information sources. The different abstraction levels are brought to a homogeneous level of detail suited for the decision makers needs. The integration of different experts advices and the amalgamation of the various preferences by means of a multi-criteria decision covers the decision-theoretical and calculating aspect of decision support: Using a model of the participants, the goals and the alternatives, IMCOD calculates the prefered solution.

3 THE *DESIGN ANALYZER* - A PROTOTYPICAL IMCOD IMPLEMENTATION

The different techniques developed in the IMCOD project were implemented in a prototypical system [8]. The so-called *Design Analyzer* supports the design of pressure tanks made of composites by performing the systematic evaluation of different design alternatives during the layout phase of the design. This section describes the modelling, the evaluation process and the decision aspects of the Design Analyzer in more detail.

3.1 Modelling Tasks in the Design Analyzer.

The Design Analyzer employs models of the applicable requirements and preferences, the available design suggestions, and the local experts integrated by the manager.

Formalization of the Requirements into the Design Specification

The intended design has to fulfill a set of requirements arising from several sources. The requirements result from the customers demands. Additional requirements arise from applicable norms and general technical guidelines and design knowledge. All these requirements can be treated in a uniform way: They are transformed into a set of design criteria as a standardized and formalized representation. Each design criterion comprises of an unambiguous description of the attribute, an interval which shall contain the value of the attribute, an indication of the optimal value of the attribute and of the tendency of the preferred deviation of the optimal value. Furthermore it is indicated whether the criterion is considered *required* or *optional*. A valid design needs to fulfill all required criteria. The optional criteria leave room for optimization, as a design does not necessarily need to fulfill all of them. In the case of an optional criterion the customers preference is indicated by a qualitative importance weight taken from a scale which offers 5 values from low to high.This representation of the design criteria facilitates the homogeneous handling of the different types of design criteria which are distinguished in general design methodologies [9]. The result of this formalization step is the *design specification.*

The formalization of the customer requirements into the design specification is supported by predefined templates of default design specifications. A template consists of a set of design criteria which might contain values in any of the elements mentioned above. The template guides the formalization by providing typical criteria and serves - by containing criteria which are usually considered common sense - as a representation of basic technical knowledge.

Modelling of the Local Experts.

The model of the local experts describes their communication interfaces (see 2.2) and their particular competences. The competences are formulated by enumerating the criteria the local expert is able to handle. For each criterion the local experts' competence is rated *high* or *low*. Experts with low competence on a criterion are considered only if no expert with high competence is available.

Initial Design Suggestions.

The design of the pressure tank starts from some rough ideas about the general outline, the material to be used, and the production methods to be applied. As there are usually several possible alternatives to reach the intended product, it is important to allow a free flow of ideas at this stage of the design process. Existing design methodologies use a brainstorming phase, where every idea about the possible design - whether new or a variant of previous solutions - is expressed and written down. The design suggestions are formally represented using the product model and additional attribute-value pairs. These design suggestions will be evaluated by the local experts.

3.2 The Evaluation Process Performed by the Design Analyzer.

The evaluation of the design suggestions uses the design specification as a guideline. For every criterion the competent local experts are selected. Handling different ontologies as noted in 2.2, the manager communicates the criteria in question and the necessary information taken from the design suggestions to the selected local experts. The experts evaluate the different suggestions and decide if and to which degree the criteria are fulfilled. To do this they may perform arbitrary operations and calculations. The experts' assessments are communicated back to the manager.

3.3 Integration of Local Expert's Assessments: A Multi-Criteria Decision.

The manager combines the evaluations of the local experts into a final ranking of the design suggestions. This is reached by calculating an extended preference model of choice.

The calculation comprises two steps: For each criterion and each design suggestion the assessments of the local experts are combined, taking into account their competences. This first step results in the assessment of a single hypothetical expert (representing the combined expertise) for each criterion. In the second step these results are integrated into a final assessment of the design suggestion, using the information about the importance of the criteria and the customers preferences. In summary the design analyzer calculates the value of the design suggestions as a weighted sum over the assessments of the local experts and the criteria. The design suggestion which receives the highest ranking value is the best one of the alternatives with respect to the requirements.

4. CONCLUSION

As an answer to the growing need for flexible information processing in modern product design we suggested the development of an intelligent design assistant. A promising approach towards a realization of this vision of a flexible and cooperating system is the integration of existing design–supporting systems under guidance and control of a central manager.

In this paper we presented an overview of the basic problems to be tackled. We focused on the communication and decision tasks of the manager. The different techniques were combined into the implementation of a prototype of this manager. The system provides decision support for the evaluation of several competing design concepts during the layout phase of the development of a pressure tank made of composites.

The approach pursued in the IMCOD project corresponds well to recent developments towards the integration of various design–supporting systems under the umbrella of an extended CAD system. The work presented here fills some voids which still remain in other approaches.

In summary IMCOD developed methods and techniques which allow for the integration of existing design–supporting systems into a homogeneous unit. The resulting design assistant offers valuable support in the area of modern product design.

Future research will focus on the handling of relevant knowledge in accordance with the workflow. Thus we continue the shift from the stand-alone problem solving system towards a cooperative paradigma of systematic knowledge management.

REFERENCES

[1] B. Bachmann, A Solution to the Semantic Unification Problem to Reuse Knowledge–Based Systems. PhD thesis, University Kaiserslautern, FB Informatik, June 1996.

[2] W.R. Swartout, R. Neches, and R. Patil. 'Knowledge sharing: Prospects and challenges'.In K. Fuchi and T. Yokoi, editors, Knowledge Building and Knowledge Sharing, pages 102–109. IOS Press, Amsterdam – Oxford – Washington, 1994.

[3] J. Owen. 'STEP. An Introduction'. Product Data Engineering. Information Geometers, 1993.

[4] B.Bachmann and F.Dridi. 'Definition of a communication layer for expert systems'. In J.G. Chen, editor, Proc. 6th Int. Conf. on AI and Expert System Applications, Houston, Texas, pages 117–123, 1994.

[5] M. Genesereth and R. Fikes. Knowledge interchange format version 3.0 reference manual. Logic Group Report Logic–92–1, Computer Science Department, Stanford University, June 1992.

[6] T. Gruber. 'A translation approach to portable ontology specification'. Knowledge Acquisition, 5(2):199–220, 1993.

[7] H.B. Enderton. A Mathematical Introduction to Logic. Academic Press, San Diego – New York – Berkeley, 1972.

[8] W. Tautges. Der DESIGN-ANALYZER – Decision Support im Designprozeß. Document D-96-03, DFKI GmbH, 1996

[9] K. Roth. Konstruieren mit Konstruktionskatalogen. Springer-Verlag, Berlin Heidelberg New York, 1982.

A KNOWLEDGE-BASED SYSTEM FOR FAULT DIAGNOSIS IN PLASTIC INJECTION MOULDING

L.H.S. Luong[1], F.T.S. Chan[2] and R.W.L. Ip[3]

[1] School of Engineering, University of South Australia, Australia
[2] Department of Industrial and Manufacturing Systems Engineering, The University of Hong Kong, Hong Kong
[3] Department of Manufacturing Engineering, City University of Hong Kong, Hong Kong

ABSTRACT

Processing of thermoplastic materials is often considered as an art rather than a science as its performance depends on the values of several processing parameters such as temperature and pressure. When a defective part occurs, it is not obvious to the operator which of the processing parameters has been incorrectly set and an experienced person with a good knowledge of thermoplastic materials and their processing requirements is needed to identify the fault and recommend the appropriate corrective actions. This paper describes the development of a prototype knowledge-based system for fault diagnosis in plastic injection moulding. The system is implemented using a database and an expert system shell, and is able to recommend qualitative corrective actions for ten defects which often occur in the processing of a range of commonly used thermoplastic materials.

KEYWORDS

Knowledge-based system, Fault Diagnosis, Injection Moulding

1. INTRODUCTION

A knowledge-based system is a computer program that uses knowledge and inference procedures to solve problems which are difficult enough to require significant expertise for the solution. The knowledge necessary to perform at such a level, plus the inference procedures used, can be thought of as a model of the expertise of the best practitioners in the field [1].

A knowledge-based system often consists of the following components:

(i) A knowledge base containing the knowledge about the application domain.

(ii) An inference engine for manipulating the stored knowledge to produce solutions to problems.

(iii) A user interface for communicating with the user. This may incorporate an explanation facility to explain to the user how the solution to a particular problem has been deduced.

A knowledge-based system without the knowledge base, ie. components (ii) and (iii) only, is known as an expert system shell. In the 1980's, applications of knowledge-based systems were often an expensive and time-consuming process as it often involved the development of all three components described above [2]. Now, with the profileration of many commercial expert system shells on the market, the application of knowledge-based system has become widespread, particularly in manufacturing. Examples include costing system [3], diagnostics of Flexible Manufacturing Systems [4], and Solid Freeform Manufacturing [5].

This paper describes the development of a knowledge-based system for fault diagnosis in plastic injection moulding. The system is used to examine faults in the moulded products and recommend appropriate corrective actions in the manufacturing process in order to overcome these problems.

2. APPLICATION DOMAIN

Applications that are computational or deterministic in nature are not good candidates for knowledge-based systems and are better handled by conventional programming. The best application candidates for knowledge-based systems are those dealing with expert heuristics for solving problems. Successful knowledge-based systems are often those that combine facts and heuristics, and thus, merge human knowledge with computer power in solving problems [6]. To be effective, a knowledge-based system must focus on a particular problem domain.

Processing of thermoplastic materials is often considered as an art rather than a science. Once the material and the required die for a particular part has been designed, quality of the moulded part is entirely dependent of the processing parameters of the injection moulding machine. Processing parameters that can be varied at the injection moulding machine include barrel temperature, mould temperature, nozzle temperature, injection pressure, back pressure and injection speed. Incorrect setting of these parameters can lead to defects in the moulded part, such as voids, burn marks, mould sticking, crazing, part distortion, short shot and shrinkage, to name only a few.

When a defective part occurs, it is not obvious to the operator which of the processing parameters has been incorrectly set. Very often, an experienced person with a good knowledge of thermoplastic materials and their processing requirements is needed to identify the fault and recommend the appropriate corrective actions for the processing parameters. For this reason, fault diagnosis in plastic injection moulding is considered a good candidate for knowledge-based system applications. This paper describes the development of such a system. The prototype system developed is able to suggest corrective actions for ten defects which often occur in the processing of a range of commonly used thermoplastic materials.

3. SYSTEM DEVELOPMENT

Fig. 1 shows the overall design of the system and the consultation algorithm. The system essentially consists of a database (PROC_PAR in Fig. 1), an inference engine, and a knowledge base. The system is implemented using DBASE IV and the VP-EXPERT shell. Knowledge representation is in the form of production rules.

The database is used to store material names and the recommended processing parameters associated with each material. Thus, each record in the database is used to store the material name and the values of the processing parameters recommended for that particular material. These parameters include maximum and minimum barrel temperature, maximum and minimum mould temperature, maximum and minimum nozzle temperature, maximum and minimum injection speed, maximum injection pressure and maximum back pressure. The range of materials available in the database is shown in Table 1.

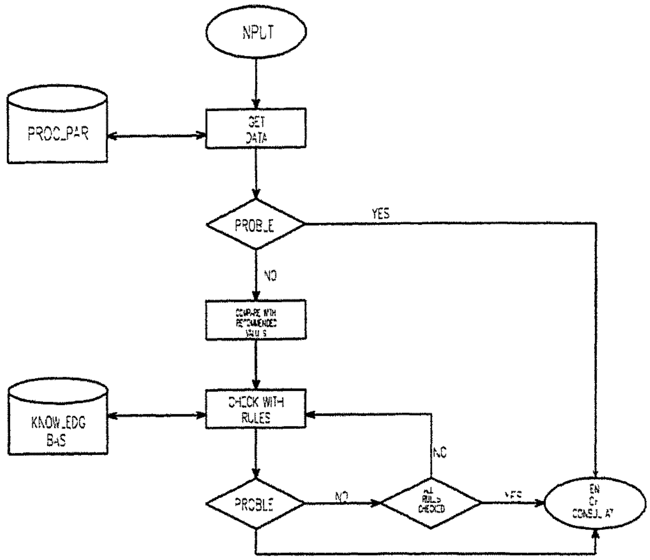


Fig. 1. Schematic diagram showing the overall design of the system and the consultation algorithm.

Common Name	**Abbreviation**
Acetals	POM
General Purpose Polystyrene	GPPS
Styrene Acrylonitrile	SAN
High Impact Polystyrene	HIPS
Polymethyl Methacrylate	PMMA
Unplasticised Polyvinyl Chloride	UPVC
Plasticised Polyvinyl Chloride	PPVC
Polycarbonate	PC
Polyethersulphone	PES
Polyenylene Oxide	PPO
Low Density Polyethylene	LDPE
High Density Polyethylene	HDPE
Polypropylene	PP
Polybutylene Terephthalate	PBT
Nylon 6	PA6
Polyphenylene Sulphide	PPS
Thermoplastic Polyurethane	TPU

Table 1. The range of materials available in the database

The knowledge required to suggest corrective actions to the ten most common defects in plastic injection moulding is represented in the form of production rules using VP-EXPERT shell. The ten processing defects and the production rules(totally 47 rules) associated with these defects are shown in Table 2, and some examples of these production rules are shown in Table 3.

Rule No.	Defects
1 - 12	Check for processing limits
13 - 15	Bubbles/ Voids
16 - 20	Burn Marks
21 - 22	Mould Sticking
23 - 27	Crazing
28 - 29	Material Degradation
30 - 32	Part Distortion
33 - 35	Flashing
36 - 39	Incorrect Shrinkage
40 - 43	Short Shot
44 - 47	Poor Welding

Table 2. List of defects and their associated production rules.

```
RULE 14
 IF defect = Bubbles AND
     injpre < (max_injpre) AND
     check[14] <> 1 AND
     problem = NO
 THEN CLS
     check[14] = 1
     action = INCREASE_INJECTION_PRESSURE
 BECAUSE "Insufficient packing pressure";

RULE 15
 IF defect = Bubbles AND
     injspd > (min_injspd) AND
     check[15] <> 1 AND
     problem = NO
 THEN CLS
     check[15] = 1
     action = DECREASE_INJECTION_SPEED
 BECAUSE "The mould filling speed is too fast";
```

Table 3. Some examples of production rules

4. SYSTEM CONSULTATION AND DISCUSSION

During consultation, the system will display a list of materials from the database and prompt the user to select the material being processed from the list. The system will also prompt the user to enter the current values of processing parameters such as barrel temperature, mould temperature, nozzle temperature and injection speed and pressure. Once the values of these parameters have been entered, the user is then asked to select the defect being observed from a displayed list.

After all the required inputs have been accepted, the system searches the database and get the recommended processing parameters for the selected material. It then proceeds to the knowledge base to check the production rules corresponding to the defect being observed (see Tables 2 and 3), and recommend appropriate action to be taken.

It should be noted that a defect on the moulded part may be due a variety of reasons. For example, the appearance of burn marks on the moulded part may be due to either high temperature, high injection speed, or high injection pressure. However, the most likely cause of burn marks is high temperature. Since the production rules are checked according to their order of appearance in the knowledge base, the rule dealing with the most likely cause would appear first. Hence in the example of the burn marks, the rule that recommends a reduction in nozzle and barrel temperatures appears before the rule for a reduction in injection speed and injection pressure. Thus during consultation the first recommended action for burn marks is reducing nozzle and barrel temperatures, and if that fails then the next course of action is to reduce injection speed and injection pressure. This method of using the order of appearance of the production rules in the knowledge base is quite subjective and based on experience. Another way of handling this problem is using a confidence factor for each likely cause of a defect. An example of a consultation session is given in Table 4.

- Please input the material being processed:
  Acetals
  General Purpose Polystyrene
  Styrene Acrylonitrile
  High Impact Polystyrene
  ....
- Please input the current barrel temperature in deg. C:
  220
- Please input the current mould temperature in deg. C:
  100
- Please input the current nozzle temperature in deg. C:
  210
- Please input the current injection pressure in MPa:
  150
- Please input the current back pressure in MPa:
  3
- Please input the current injection speed (Fast, Medium, Slow):
  Fast
- Please input the defect being observed:
  Bubbles/ Voids
  Burn Marks
  Mould Sticking
  Crazing
  ....

Output - Recommended actions listed in the order of priority:

1. Reduce barrel temperature
2. Reduce back pressure
3. Reduce nozzle temperature
4. Reduce injection speed.

Table 4. Example of a consultation session

5. CONCLUSION

A prototype knowledge based system has been developed for fault diagnosis in plastic injection moulding process. The system is able to recommend qualitative corrective actions for ten defects which often occur in the processing of a range of commonly used thermoplastic materials. The system has been implemented using a database and an expert system shell. The usefulness of the system can be enhanced by expanding the range of materials in the database and the range of production rules dealing with various defects. At present the system is only able to make qualitative recommended action such as increasing or decreasing temperature. Further work may also include some quantitative recommendations.

6. REFERENCES

1. Feigenbaum, E.A., "Expert Systems in the 1980s" in: Machine Intelligence, A. Bond (ed.), Pergamon, Oxford, 1981.

2. Luong, L.H.S. "Applications of Expert Systems in Manufacturing: A Case Study", J. of Computers in Industry, vol. 18, pp. 193-198, 1992.

3. Chin, K.S. and Pun, K.F., "Expert Systems for Concurrent Engineering: An Expert Injection Mould Costing System", Proc. Int. Conf. on Data and Knowledge Systems for Manufacturing and Engineering, Hong Kong, May 2-4, 1994, The Chinese University of Hong Kong, pp. 174-179, 1994.

4. Torvinen, S. and Milne, R., "Increasing the Availability of Flexible Manufacturing Systems with Multimedia Supported Diagnostic Expert System", Proc. 3rd Int. Conf. on Computer Integrated Manufacturing, Singapore, July 11-14, 1995, Gintic Institute of Manufacturing Technology, pp. 1365-1372, 1995.

5. Narayanan, V., Chua, C.K. and Ang, B., "A Selection Expert System for Solid Freeform Manufacturing", Proc. 3rd Int. Conf. on Computer Integrated Manufacturing, Singapore, July 11-14, 1995, Gintic Institute of Manufacturing Technology, pp. 1416-1426, 1995.

6. Badiru, A.B., "Expert Systems Applications in Engineering and Manufacturing", Prentice Hall, Englewood, New Jersey, 1992.

FUZZY GENETIC ALGORITHMS FOR OPERATION SEQUENCING

T. N. WONG and L.C.F.CHAN
Department of Industrial and Manufacturing Systems Engineering
The University of Hong Kong
Pokfulam Road, Hong Kong

ABSTRACT

This paper describes a fuzzy approach for solving the operation sequencing problem in process planning. The reason for using fuzzy number and fuzzy arithmetic in representing quantitative information is investigated. Then, cost analysis of a complete machining cycle is performed and the mathematical model of the operation sequencing problem is formulated. Genetic algorithms with fuzzy numbers and fuzzy arithmetic are developed to solve this global sequencing problem. New chromosome representation scheme, mutation and crossover operators, and evaluation functions are designed to solve this sequencing problem.

KEYWORDS
Process sequencing, Genetic algorithms, Fuzzy arithmatic

1 INTRODUCTION

Process planning is an important manufacturing activity that generates a set of manufacturing instructions in order to convert a workpiece from its initial states into the final form. Operation sequencing is one of the major sub-tasks of process planning. Methodologies of solving operation sequencing can be cataloged into two main approaches: the knowledge-based approach and mathematical models.

In the knowledge-based approach, a set of sequencing rules are defined to order to generate a feasible operation sequence. Examples are the works by Wang and Wysk[1], Davies and Darbyshire[2], Wong and Siu[3]. The advantage of this approach is that it is easily implemented and the calculation time is short. However, the drawback is that the solution is only a feasible one and far from optimal.

The other approach is based on mathematical models. For instance, the operation sequencing problem can be formulated as a Travelling Salesman Problem, a well-known NP-complete problem. An example is the work by Lin[4]. Using this approach, the solution is usually optimal or near-optimal. However, deal to the difficulties in achieving precise and reliable measurements, some parameters are very difficult to obtain and are vaguely known.

In this paper, a fuzzy genetic algorithm approach is proposed in order to overcome the drawbacks of the traditional approaches. In our approach, all the quantitative information is expressed in fuzzy number and the genetic algorithm with fuzzy arithmetic operators are employed in generating the near-optimal global operation sequence.

2. REASONS FOR USING FUZZY NUMBERS IN PROCESS PLANING

Traditional process planning systems generate process plans based on the assumption that all the parameters can be characterized by a precise and crisp value. There are two main reasons for the use of fuzzy numbers in calculating the machining cost:

(a) Because of the uncertainties and inaccuracy of data, it is very difficult to get a precise measure of some of the manufacturing cost factors, such as the machining time, setup time and etc.

(b) Many of the so-called precise manufacturing data can only be applied to some of the very special and simplified case. Instead of having a different sets of precise manufacturing data for each different case, we can use the fuzzy number to simulate a general condition.

3 MATHEMATICAL MODEL OF OPERATION SEQUENCING PROBLEM

3.1 Cost Analysis

A machining cycle consists of two types of activities. The first one is the metal removing activity called machining operation or operation. The second one is the preparation activities which prepare all the necessary manufacturing resources for the operation, such as material traveling, part loading\unloading, part clamping\un-clamping, tool changing, fixture setup and etc. In general case, if the number of operations are n and the operation sequence is $O_1, O_2, ..., O_i, ... O_n$ then the manufacturing cost incurred will be:

$$Cost(O_1, O_2, \quad , O_n) = \sum_{i=1}^{n} M(O_i) + P(O_S, O_0) + \sum_{i=0}^{n-1} P(O_i, O_{i+1}) + P(O_n, O_E) \tag{1}$$

$$\begin{aligned} P(O_S,O_i) &= CT(O_i)+CM(O_i)+ CTS(O_i)+CFS(O_i)+CC(O_i) \\ P(O_i,O_j) &= CU(O_i)+CT(O_j)+CM(O_j)+CTS(O_j)+CFS(O_j)+CC(O_j) \\ P(O_j,O_E) &= CU(O_j)+CT(O_j) \end{aligned} \tag{2}$$

where

- O_S be the starting condition before processing any operation unit,
- O_E be the ending condition before processing any operation unit,
- $M(O_i)$ be the operation cost for operation O_i.
- $P(O_i,O_j)$ be the preparation cost for operation units O_j after operation units O_i,
- $CTR(O_1)$is the cost incurred from material traveling to the machine performing O_1
- $CM(O_1)$is the cost incurred from machine setup for O_1
- $CS(O_1)$is the cost incurred from material traveling to the machine performing O_1
- $CC(O_1)$ and $CU(O_1)$is the cost of clamping and un-clamping the part on fixture for O_1
- $CTS(O_1)$and $CFS(O_1)$ are the cost of tool setup and fixture setup for operation O_1

From equation (1), we find that the manufacturing costs of the metal removing process can be divided into two types: operation cost, $M(O_i)$ for each operation O_i and preparation cost, $P(O_i,O_j)$ between two consecutive operations O_i and O_j.

It is observed that the operation cost is operation-sequence independent, that is, the cost is independent of the operation sequence of the process plan. The total operation cost is simply the summation of all the individual operation costs in the process plan whatever the sequence is. On the contrary, the preparation cost of an operation is operation-sequence dependence. The preparation costs depends on the difference between the manufacturing resources allocated in two consecutive operation units. For example, there will be no part load/unload and material handling cost if two consecutive operations are performed on the same machine with the same setup. Therefore, the preparation cost can be modified into the following equations :

$$\begin{aligned} P(O_S,O_i) &= k_2*CTR(O_S,O_i)+k_2*CM(O_i)+k_3*CTS(O_i)+k_1*CFS(O_i)+k_1*CC(O_i) \\ P(O_i,O_j) &= k_1*CU(O_i)+k_2*CTR(O_i,O_j)+k_2*CM(O_j)+k_3*CTS(O_j)+k_1*CFS(O_j)+k_1*CC(O_j) \\ P(O_j,O_E) &= k_1*CU(O_j)+k_2*CTR(O_j) \end{aligned} \tag{3}$$

where

$k_1 = 0$ if Same Machine and fixture setup for O_i and O_{i+1}
$= 1$ else
$k_2 = 0$ if Same Machine for O_i and O_{i+1}
$= 1$ else
$k_3 = 0$ if Same Machine and cutting tools for O_i and O_{i+1}
$= 1$ else

3.2 Problem formulation

Suppose for a design part, there is a set of feature elements denoted by $\{F_i\}$, i=1,2,...,m. Each feature element F_i corresponds to a local sequence of operation units denoted by $(O_{i1}, O_{i2}, ..., O_{in_i})$ where n_i is the total number of operation units for feature F_i.

Let $\{O_j\}$ j=1,2,...,K be the set of operation units that can process all the features elements of a design part where $K=n_1+n_2+...+n_m$. The planning objective of the operation sequencing problem is to select a global operation sequence O: $<O_{\pi(1)}, O_{\pi(2)}, ..., O_{\pi(K)}>$ among the operation units set $\{O_1 ....., O_K\}$, such that the manufacturing cost is at a minimum and under the constraint of the local precedence order of operation units for each feature element :

$$\sum_{i=1}^{K} M(O_{\pi(i)}) + P(O_S, O_{\pi(0)}) + [\sum_{i=0}^{K-1} P(O_{\pi(i)}, O_{\pi(i+1)})] + P(O_{\pi(K)}, O_E) = \text{minimum} \quad (4)$$

It is observed that the preparation cost is operation sequence dependence while the operation cost is sequence independent. The value of $\Sigma M(O_{\pi(i)})$ is always the same whatever the sequence is, therefore the objective function can be reduced to:

$$P(O_S, O_{\pi(0)}) + [\sum_{i=0}^{K-1} P(O_{\pi(i)}, O_{\pi(i+1)})] + P(O_{\pi(K)}, O_E) = \text{minimum} \quad (5)$$

4 GENETIC ALGORITHM FOR OPERATION SEQUENCING PROBLEM

In our research, genetic algorithm is adopted to solve this operation sequencing problem. Quantitative information is expressed in fuzzy number and the evaluation is performed by fuzzy arithmetic operators. New crossover and mutation operators are also designed in order to generate a feasible global OMTF sequence.

4.1 Chromosome representation

The chromosome structure represents the global sequence of the operation units for the feature cluster. In our research, the path representation approach is employed in chromosome representation. Path representation is the most natural representation of a sequence. Suppose that the candidate global operation sequence is

O: $< O_{\pi(1)}, ..., O_{\pi(K)} >$

then the chromosome representation for this candidate sequence will simply be

O: $(O_{\pi(1)}, O_{\pi(2)}, ..., O_{\pi(K)})$

4.2 Initialization

Initial procedure constructs a population of chromosomes for the first iteration. However, since the local precedence order of the operation units for each feature elements within the feature cluster must be preserved, therefore, this initial chromosome pool can not be constructed by simple random generation.

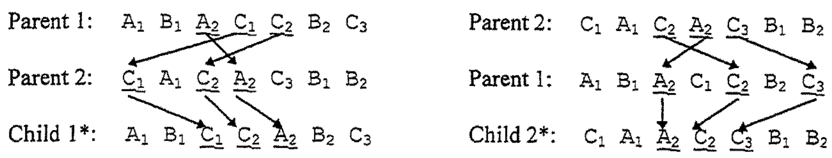
Among the set of operation units having highest priority, we randomly select one as a gene. The selected operation units will be deleted from the its local operation sequence and a new set of operation unit having highest priority will be exposed. Then randomly select another operation unit as the next gene among the new set of operations having highest priority. This procedure will be repeated until all the operation units in the local operation sequence have been possessed.

Suppose there are three feature elements and the corresponding local operation sequence be $(A_1\ A_2)$ $(B_1\ B_2)$ $(C_1\ C_2\ C_3)$, the operation units having highest priority are A_1, B_1 and C_1. If A_1 is selected as the first gene, then the next highest priority operation units will be A_2, B_1 and C_1. If B_1 is selected as the next gene, then the next highest priority operation units will be A_2, B_2 and C_1.

4.3 Crossover

In order to produce feasible offspring, two parents are randomly selected from the population, one parent act as a Crossover chromosome and the other one will act as a Reference chromosome. A random segment will be selected from the Crossover chromosome and those gene appeared in the selected segment will be highlighted in the Reference chromosome. Then the gene within the selected segment of the Crossover chromosome will be rearranged according to the order of the highlighted gene of the Reference chromosome and a child chromosome will be produced. The role of these two parents will then exchange in order to produce another child chromosome. The following example will illustrate the mechanism of our crossover operation:

Suppose there are three feature elements and the corresponding local operation sequence be $(A_1\ A_2)\ (B_1\ B_2)\ (C_1\ C_2\ C_3)$, then



Parent 1: A_1 B_1 A_2 C_1 C_2 B_2 C_3 ⟶ Child 1*: A_1 B_1 C_1 C_2 A_2 B_2 C_3
Parent 2: C_1 A_1 C_2 A_2 C_3 B_1 B_2 ⟶ Child 2*: C_1 A_1 A_2 C_2 C_3 B_1 B_2

4.4 Mutation

From the gene pool, randomly select a piece of chromosome as Parent chromosome. Randomly select a gene on the Parent chromosome as the Selected Gene. Locate the Bounded Gene on the Parent chromosome which are the preceding and succeeding operation unit in the local operation sequence of the selected gene. The segment bounded by these two genes are called the Candidate Segment. This Selected gene can exchange with another gene within the candidate segment and a child chromosome can be produced. Chromosome performed in this way can guarantee to generate a feasible offspring. The following illustrates the procedure of our mutation operation.

Parent chromosome	:	A_1	C_1	A_2	B_1	C_2	B_2	C_3	Selected Gene : C_2
Bounded Gene for C_2	:	C_1 and C_3							
Candidate Segment	:	A_1	C_1	A_2	B_1	C_2	B_2	C_3	
Child *	:	A_1	C_1	C_2	B_1	A_2	B_2	C_3	

4.5 Evaluation and selection

Each piece of chromosome which represents the structure of a valid operation sequence in the population pool is then evaluated by the objective function. A fitness value is assigned to each piece of chromosome. Let $O_i : (O_{i1}, O_{i2}, ...O_{iK})$ denote the i-th chromosome of the population, O be the starting machining condition before processing the feature cluster, the evaluation function $C_i(O_i)$. A particular global operation sequence is the cost function as given in equation (1).

Since the evaluation cost value is a triangular fuzzy number, it is impossible to directly compare with each other and generate a fitness value. In our research, we employ the approach proposed by Kaufmann [5, 6]. He proposed a Removal function which map a fuzzy number to a real number for comparison. For triangular fuzzy numbers A : (a, b, c), this ranking function will be as follows :

$$R(A) = \frac{a + 2b + c}{4} \tag{6}$$

For each chromosome, we calculate the Removal number for their total fuzzy cost by the removal function $R(C_i(O_i))$. Because this is an minimization problem, chromosome having smaller fuzzy total cost should have a higher probability to be selected and higher fitness value. This can be handled easily by simply taking reciprocal of the removal number. Thus we have:

$$\text{eval}(O_i) = \frac{1}{R(C_i(O_i)} \tag{7}$$

where $\text{eval}(O_i)$ is the fitness function for the i-th chromosome. With this fitness function, we can reproduce a new population for the next iteration

5 Example

The algorithm will be illustrated by the following example. Suppose we have a prismatic part with a through hole, a blind hole and a slot. The local operation sequence is centre-drilling-> drilling-> reaming for through hole, centre-drilling -> drilling for blind hole and milling for slot. The resources allocated as below:

Feature	Operation	Machine	Cutting tools	Fixturing
Through Hole	O_{11}centre-drilling	Drilling	3mm dia drill	Z-direction
	O_{12} drilling	Drilling	5mm dia drill	Z-direction
	O_{13} reaming	Drilling	5mm dia reamer	Z-direction
Through Hole	O_{21} centre-drilling	Drilling	3mm dia drill	X-direction
	O_{22} drilling	Drilling	6mm dia drill	X-direction
	O_{23} reaming	Drilling	6mm dia reamer	X-direction
Blind Hole	O_{31} centre-drilling	Drilling	3mm dia drill	Y-direction
	O_{32} drilling	Drilling	10mm dia drill	Y-direction
	O_{33} drilling	Drilling	10mm dia reamer	Y-direction
Blind Hole	O_{41} centre-drilling	Drilling	3mm dia drill	Z-direction
	O_{42} drilling	Drilling	12mm dia drill	Z-direction
	O_{43} drilling	Drilling	12mm dia ream	Z-direction
Slot	O_{51} milling	Milling	15mm dia end mill	X-direction
Through Hole	O_{61} centre-drilling	Drilling	3mm dia drill	X-direction
	O_{62} drilling	Drilling	5mm dia drill	X-direction
Through Hole	O_{71} centre-drilling	Drilling	3mm dia drill	Y-direction
	O_{72} drilling	Drilling	6mm dia drill	Y-direction

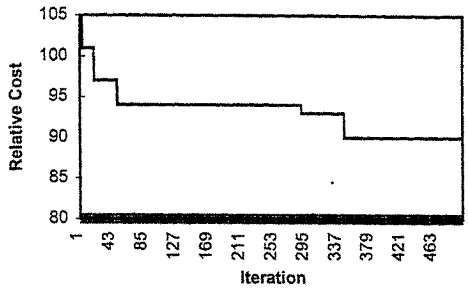
Suppose we have the following relative cost for this example,

Part Unclamping/clamping	: (1,2,3)	Milling tools setup	: (5,7,9)
Part traveling	: (2,3,4)	Drilling machine setup	: (6,8,10)
Drilling tools setup	: (2,3,4)	Milling machine setup	: (7,9,11)
Reaming tools setup	: (3,5,7)	Fixture setup	: (1,2,3)

In our implementation, the following parameters will be used:

Population size	: 100
Mutation Probability	: 0.7
Crossover Probability	: 0.7
Iteration	: 500

The optimal operation sequence selected after 100 iteration will be
(O_{51}, O_{21}, O_{61}, O_{22}, O_{62}, O_{23}, O_{41}, O_{11}, O_{12}, O_{42}, O_{43}, O_{13}, O_{31}, O_{71}, O_{32}, O_{72}, O_{33}) with the defuzzified relative cost equals 90.



6 Conclusion

In conclusion, a genetic algorithm with fuzzy arithmetic is proposed to solve the operation sequencing problem, in which, all the cost factors are expressed in fuzzy number. In order not to violate the local operation sequence constraints, a new chromosome representation scheme, mutation and crossover operators, and evaluation function are designed. And, finally, the new method is verified by an example.

7 Reference

1. Wang, H. P. and Wysk, R. A, "A knowledge-based approach for automated process planning," International Journal of Production Research," 26(6), pp. 999-1014, 1988.
2. Davies, B.J., and Daryshire, I.L., "The use of expert systems in process planning", Annals of the CIRP, 33, pp. 303-306.
3. Wong, T.N. and Siu, S.L. "A knowledge-based approach to automated machining process selection and sequencing", International Journal of Production Research, 33(12), pp. 3465-3484, 1995.
4. Lin, Chun-Wei Remen, Computer-aided alternative process planning in a dynamically changing manufacturing environment, Chapter 4, The Pennsylvania State University, May 1993.
5. Kaufmann, A and Gupta, M. M., Fuzzy mathematical models in engineering and management science., Chapter 1, North-Holland, Amsterdam,1988.
6. Kaufmann, A., Introduction to fuzzy arithmetic : theory and applications., Chapter 1, Van Nostrand Reinhold, New York, 1985.

ALGORITHMIC MODELLING OF A JOB-SHOP TYPE TURNING CELL

P Zhang and R Bell
Department of Manufacturing Engineering
Loughborough University
United Kingdom

ABSTRACT

This paper presents the implementation of an algorithmic model to study and evaluate a job-shop type industrial installation. Data has been obtained for a particular job shop and the model has been used to illustrate the value of using a tool orientated approach to the assessment of both the design and operation of the installation.

KEYWORDS

Algorithmic modelling, Tool management, Job shop manufacture.

1 INTRODUCTION

Batch manufacturing systems for cylindrical components feature frequent set-up and tool changing operations and interaction with intermediate processing at other workstations. Structured computer modelling tools can be used to forecast system performance and as design tools to evaluate alternative system designs. Research and implementation studies have been carried out in the area of simulation and modelling [1], scheduling and operation strategies [2, 3]. An algorithmic model has been built by the authors to evaluate the merits of different system designs, including operation and tooling strategies. The model has the ability to record considerable amounts of user specific data on the operation of turning systems, tool and tool flow systems. The comprehensive use of the algorithmic model provides outputs which can be employed to improve the overall tool flow system performance The model can be used to represent the extended performance of a particular installation, giving valuable assessments of scheduling and work routing rules and tool management strategies. The required levels of operator support can also be deduced.

This paper presents the results of a design study to evaluate the performance of a shop which consists of five CNC turning machines; a number of alternative designs are evaluated with the aid of the algorithmic model. The cell features extensive operator involvement, frequent interactions with other machines, and a long set-up time. The case study presented here illustrates how the computer model, with its production scheduling algorithms, tooling algorithms, and manning requirement algorithms, can be employed to evaluate the performance of an in-operation job-shop type turning cell and to examine alternative system designs and operating strategies. The emphasis of the algorithmic model is biased towards modelling the tooling provision. However, it also serves as a system design aid and a fast and effective insight into cell performance, which can be further complemented by the use of conventional discrete event simulation.

2 SHOP DESCRIPTION

The algorithmic model is applied to a particular industrial instance, ie the "turbine shop", which is a workshop used solely for bar and chuck type components for turbine assemblies. It is centred on a five-CNC lathe turning cell supported by a number of NC and conventional machines (Figure 1). The five-machine CNC turning cell has been modelled with the inclusion of tooling back-up by the tool set-up area – the de facto Central Tool Store (CTS) for the shop. The interaction between the CNC turning cell and the other machines has been fully represented with regard to the part flow out and the CNC turning cell. The CNC turning cell consists of three types (hence groups) of CNC lathes. Machines 1, 2 and 3 (type/group I) have the ability to machine bars of up to 65 mm and chuck parts of up to approximately 200 mm in diameter. The crown turret, with 12 tool stations, for either internal or external machining. The tool driving attachment of the turret allows up to six live tools to be accommodated. Machine 4 (type/group II) and Machine 5 (type/group III) have the ability to machine bar type components of up to 42 and 65 mm diameter respectively. Both of them can turn chuck parts of up to 200 mm in diameter. External and internal turning tools can be arranged in their 14 station turret in any order. With C-axis control facility and the tool driving mechanism attached to the turret. Machines 4 and 5 can carry secondary operations by accommodating up to seven live tools.

The case study encompasses the production of eleven 11 different part types. The part spectrum includes turbine wheel, exducer, compressor impeller, impeller fan axial, and exducer for cooling turbine. The processing of these components inside the CNC turning cell is assigned to one of the three machine groups according to the machine type required. The tool operations performed in the CNC cell range from 2 to 14. All the components require machining inside the CNC turning cell and operations outside the CNC cell, eg milling, turning, grinding, deburring, inspection and balancing.

A typical part routing is given in Figure 2. Before entering the CNC turning cell, the part requires operations on the NC lathe. Then 11 tool operations including turning, boring, and central drilling will be performed on Machine 1, after which the part leaves the CNC turning cell for outside operations on the NC mill and NC lathe. The batch will then enter the CNC turning cell again for processing on Machine 2 and will leave the CNC turning cell for deburr, inspection and final boring operations (Figure 2).

3 PRODUCTION SCHEDULING STRATEGIES

The production scheduling algorithm adopts a three step approach, ie part launching, work load assignment, and batch sequencing [4]. Four part types require initial operations before entering the CNC turning cell and six part types require exit and re-entry to the CNC cell before the processing in the turbine cell is completed. The launching rule adopted was that a batch can only enter the CNC turning cell when its previous operations have been finished out of the cell. The fact that three type I turning centres are pooled into a group indicates that the workload for the three machines should be balanced. Two alternative rules have been implemented for the assignment of work load to individual machines. *Workload Assignment Rule 1*: The machine with the least workload of the group will be assigned the batch that has the maximum work-content that can be performed by the machine. *Workload Assignment Rule 2*: The machine with the least workload of the group will be assigned the batch that has the minimum work-content that can be performed by the machine.

Assigned batches have to be sequenced by one of the following rules. *Sequencing Rule 1*: When a machine is available, the ready batch which requires the longest time interval between its exit and re-entry later during its processing will be sequenced first. For batches with the same amount of outside operation time, the one with the shortest processing time will be sequenced first. *Sequencing Rule 2*: No priority is given to batches that require exit/re-entry. Shortest Processing Time (SPT) is employed for sequencing: ie the batch which requires the shortest processing time among all the batches waiting for an available machine will be sequenced first. *Sequencing Rule 3*: When a machine is available, all the batches waiting for the machine are listed. The batch which requires the least number of tools to be exchanged to the machine will be sequenced first. This is implemented with the *differential kitting* strategy which is described below. It is aimed at maximising tool sharing between successive tool kits. Of these three sequencing rules, the first two are the function of the *External Production Scheduler*; and Sequencing Rule 3 is the function of the *Next-step Scheduler*. Six possible production strategies are obtained from the combination of the two *workload assignment rules* and the three *sequencing strategy rules*:

Scheduling Strategy 1: workload assignment rule 1 combined with *sequencing rule 1;*
Scheduling Strategy 2: workload assignment rule 2 combined with *sequencing rule 1;*
Scheduling Strategy 3: workload assignment rule 2 combined with *sequencing rule 2;*
Scheduling Strategy 4: workload assignment rule 1 combined with *sequencing rule 2;*
Scheduling Strategy 5: workload assignment rule 1 combined with *sequencing rule 3;*
Scheduling Strategy 6: workload assignment rule 2 combined with *sequencing rule 3.*

4 TOOLING AND TOOLING STRATEGY

Three tooling strategies have been implemented to suit different system designs and level of turning system automation, and the tooling requirements associated with each of these strategies is a significant parameter [4]. *Kitting*: Tool kits are issued to a machine to accompany each batch. When the processing of a batch is finished, the tool kit in the machine will be exchanged with the new kit for the next batch. The used kit will then be transferred back to the tool refurbishing and presetting area. No tool sharing between kits is permitted. During the tooling set-up period for each batch, the operator is required to prepare the tool kits, unload tools from the previous kit, and mount the new tool kit on the machine tool turret. It is specified that 15 minutes is required for each tool preparation and exchange. The machine is not in operation during this tooling set-up time. *Differential Kitting*: Tools are assigned in kits for each batch, but common tools can be shared between successive tool kits. A tool kit contains only tools that are not available in the machine. Tool savings can be achieved by using the tool life management capability of the CNC controller. This is possible on the more highly automated turning systems. *Complete Tool Magazine Exchange*: Tools for a production period are stored in the magazine up to the capacity of the magazine. When the machining assignment is completed the complete magazine of tools is replaced.

5 SYSTEM PERFORMANCE EVALUATION CRITERIA

The following definitions of machine activity times have been employed for the purpose of system performance evaluation. The *total processing time* of a machine is defined as the interval between the time the machine starts processing the first batch and the time that the last batch is finished. It is broken down into *cutting time*, which is the time a machine is actually doing the metal removal; *part set-up time*, which is the time for batch set-up and part loading/unloading; *tooling set-up time*, which is the time that a machine spends for tool presetting, exchanging, and turret indexing; and the *machine idle time*, which is the sum of the time that a machine spends during its total processing period waiting for batches to finish their previous operations before they can be sequenced to the machine. The longest *total processing time* of individual machines is defined as the *throughput time* of the turning cell. The *total idle time* of the cell is defined as the sum of *machine idle time* of all machines in the cell. The *throughput time* is used as the most important criterion for cell performance evaluation. The *total idle time* is meaningful only when examining the waste of the cost of manpower and equipment.

6 MODELLING OF THE CELL OPTIONS

The turning cell, as depicted in Figure 1, was used to conduct experiments with four scheduling strategies. The cell currently operates with a kitting tool strategy. Fifteen minutes is specified as the time required to prepare and exchange a tool in the machine turret. A machine is not in operation during part loading/unloading and tool exchange. The previous studies showed that Machines 4 and 5 were under-utilised. These two machines are of a similar specification and, in this particular case, they have the same processing capability for the assigned part types. It was therefore decided to eliminate Machine 5 from the cell layout by shifting the production requirement of that machine to Machine 4, and to examine the consequences. The result proved that this was an appropriate approach for the 11 component types and their associated processing information.

The case study was extended to include an alternative system design where the turning cell was assumed to consist of two highly automated CNC turning systems [5]. This extension allowed a critical comparison with the initial studies, which showed that a very large percentage of the processing time had been engaged in setting-up and that the actual cutting time of each machine was very low. Each batch requires different chuck jaws to fulfil the processing requirements and the high accuracy required results in a lengthy part set-up time; the tools required are assembled and preset by the machine operator. The machine is idle during the tooling set-up periods. It is therefore appropriate to study the possibility of employing highly automated machines with features which include tool magazine support and automated chuck jaw changing. For these two highly automated machines, tooling strategies, *differential kitting*, and *complete magazine exchange* can be implemented. In this cell, it is assumed that tools are assembled, broken down and preset in the tool presetting area by dedicated tool preparation staff. It is assumed that 30 minutes is required for tool magazine exchange and 15 minutes for tool kit exchanging. A machine is assumed to be not in operation when its tool magazine is being serviced. A highly automated machine of this type has been fully evaluated using the algorithmic model [4]. The conditions studied with the aid of the model are summarised in Table 1 and the results are presented in Table 2. The manning patterns obtained for a particular case, ie the two-machine cell, are depicted in Figures 3 and 4 to illustrate the output provided by the model.

7 CONCLUSIONS

The case study has shown how the scheduling strategies and the view taken of tool management affect both the system performance and tooling requirement. The scheduling strategies which consider the turning cell production environment, ie the part exit/re-entry, have resulted in significantly shorter throughput times than others. The effectiveness and performance of the scheduling and tooling strategies are very much production requirement and tooling system dependent. The density of manning requirement is affected to a significant extent by the tool issue strategies. The break down of tool assemblies into tool components has been an efficient forecasting tool for central tool store inventory control, it has also proved that altering the permissible tool life utilisation limit affects the tool requirement. The relationship between the assumed tool life and the tool requirement is system dependent, and no formal relationship could be established between them. The case study also showed how the algorithmic model could be employed to assess the effectiveness of the turning cell specification, both in respect of the number of machines used and the level of automation they employ.

8 ACKNOWLEDGEMENTS

The authors wish to thank EPSRC/DIPG for funding the project. Thanks are also extended to the following companies for their collaboration: APV Baker, British Aerospace, Cross International, GEC FAST, Lucas Industries, Rolls-Royce, Matrix-Churchill, and Istel. We also thank Normalair-Garrett Ltd (NGL) and Traub AG for providing the data used in this research. The research workers in the Department are also acknowledged, especially colleagues in the FMS Research Group and Tool Management Group, with whom stimulating and helpful discussions have taken place.

9 REFERENCES

[1] Alting L, Bilberg A and Larsen N E, "Extend Applications of Simulation in Manufacturing Systems", Annals of the CIRP, Vol 37/1/1988, pp 417-420, 1988.

[2] Abdin M F, "Solution of Scheduling Problems of Job-Shop Type FMS with Alternative Machine Tools", Proceedings of the 8th Annual Conference on Computers and Industrial Engineering, Vol 11, Nos 1-4, pp 241-245, 1986.

[3] Seliger G, Viehweger B and Wieneke B, "Decision Support for Planning Flexible Manufacturing Systems", International Trends in Manufacturing Technology Simulation – Application in Manufacturing, pp 149-162, 1986.

[4] Zhang P, Tool Flow Management in Batch Manufacturing Systems for Cylindrical Components, PhD Thesis, Loughborough University of Technology, UK, 1989.

[5] Ganiyusufoglu O S, "Building Automated Machine Cells – A Step by Step Approach", Flexible Manufacturing for Small to Medium Enterprises – A European Conference, K McGuigan (Editor), pp 43-75, IFS Publications/Springer-Verlag, Dublin, 1988.

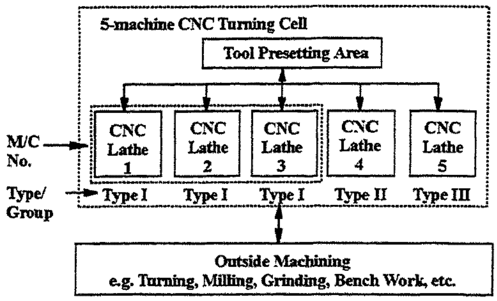


Figure 1: CNC Turning Cell Configuration

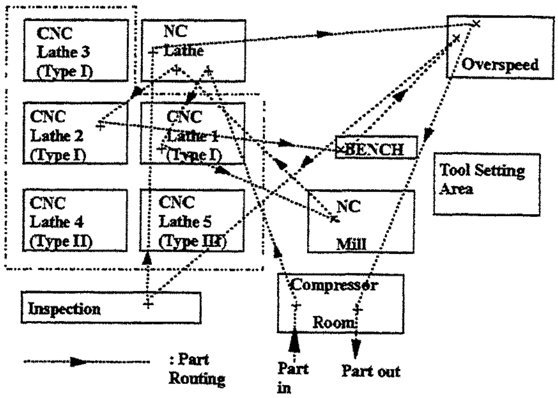


Figure 2: Turbine Shop Layout with a Typical Part Routing Example

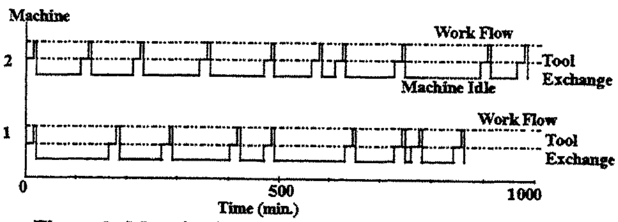


Figure 3: Manning Pattern- 2 Machine Cell with Different Tooling Strategy

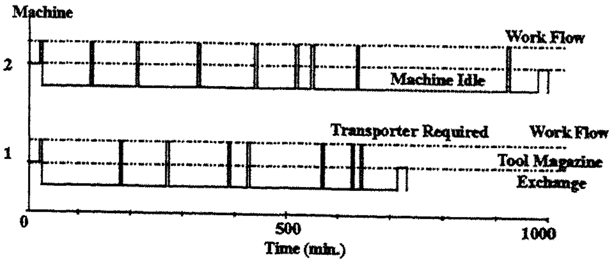


Figure 4: Manning Pattern- 2 Machine Cell with Complete Magazine Exchange Tooling Strategy

Data Specification & Cell Design	Tooling Strategy	Scheduling Strategy	Comments
- Original part data - 5-machine cell design - 11 part types - 50% permissible tool life	1. Kitting	1	- Initial installation with alternative scheduling strategies
		2	
		3	
		4	
- Original part data - 4-machine cell design - 50% permissible tool life	1. Kitting	1	- Machine 4 & 5 under-utilised - Machine 4 & 5 of same type - What-if machine 5 is eliminated
		2	
- 2-machine automated cell - Automatic Chuck Jaw Changing (ACJC) machine - Tool magazine added - Reduced set-up time	2. Differential Kitting	1	- Differential kitting proved tooling saving over kitting - Differential kitting applicable when tool magazine equipped - Tooling Strategy 3 reduces tool exchanging interfere
		2	
	3. Complete Magazine Exchange	1	
		2	
- Original part data - 4-machine cell design - Increased/ decreased permissible tool life limit	1. Kitting	2	- Permissible tool life = 80% - Scheduling Strategy 2 performs better
		2	- Permissible tool life = 65%
- 2-machine automated cell - Increased/ decreased permissible tool life limit - Supplied part data reduced set-up time	2. Differential Kitting	2	- Permissible tool life = 80%
		2	- Permissible tool life = 65%

Table 1: Overview of Modelling Experiments

Data Specification & Cell Design	Tooling Strategy	Scheduling Strategy	Throughput Time (min.)	Tool/Tool Component Requirement			
				Tools	Inserts	Shanks	Holders
- Original 5-machine cell	1	1	1436	111	74	47	66
		2	1346	111	84	56	71
		3	1812	111	77	54	66
		4	1829	111	78	56	79
- 4-machine cell	1	1	1436	111	76	48	64
		2	1346	111	87	57	70
- 2-machine highly automated cell	2	1	965	88	74	50	71
		2	965	89	70	46	55
	3	1	980	95	95	68	95
		2	980	97	97	70	97
- 4-machine cell - Relaxed Tool Life Limit	1	2	1293	99	75	47	60
		2	1308	102	78	49	62
- 2-machine highly automated cell - Relaxed Tool Life Limit	2	2	964	71	58	35	49
		2	964	80	66	41	56

Table 2: Summary of Outputs

THE STRUCTURE AND IMPLEMENTATION OF A TASK-CENTRED MANUFACTURING INFORMATION SYSTEM FOR SMALL-TO-MEDIUM-SIZED ENGINEERING COMPANIES.

Bin Wu & Ray Ellis
School of Industrial and Manufacturing Science,
Cranfield University
Bedfordshire
MK43 OAL, England UK

ABSTRACT

The structure of a typical manufacturing information system (MIS) has followed a traditional approach to software development. Such a system is usually built around a complex database and accesses the appropriate database as and when the user requests. Until recently this approach has been quite satisfactory for stand alone applications. However, from a user's point of view, one of the problems with such an approach is the difficulties associated with adaptability, extensibility and compatibility with other information technology (IT) tools. Ideally companies do not wish to approach the job in hand with the restriction of a single IT tool/database, and have began to demand the ability to achieve higher levels of integration among the software tools that they use. This paper reports the development and implementation of a task-centred MIS for small-to-medium-sized engineering companies. The system aims to provide a working environment which will enable the user to collect, select and present manufacturing data according to the specific needs of a particular company and their current work. The system developed, known as KIDS, has been given the 1996 UK Machinery Award for Innovation in Production Engineering, for being "the most innovative application of computer technology in the manufacturing environment"

KEYWORDS

Manufacturing Information System, CAD/CAM, Data Transfer, Task-Centred Integration, Object-Orientated Database.

1. INTRODUCTION

Demands on manufacturing industry to provide flexibility and reduce costs have put pressures on manufacturing companies to improve productivity. These demands, coupled with computer hardware and software advances, have encouraged MIS development. As a result the role and importance of MIS within the manufacturing environment have changed dramatically in recent years.

However, the application of a traditional complex database as an information system solution has many restrictions. The initial design of the database must be vary carefully considered [1, 2]. The design criteria requires the identification of the functions to be performed and the data required for each of these functions, so that a logical design can be made. The way in which such a database is structured and organised has a profound effect on the way in which information can be delivered, and it also places restrictions or limitations on future extendibility and adaptability if optimum design performance is to be maintained. In addition, traditional information databases were generally designed to store homogeneous, structured numeric and character data, and were not designed to cater for the change in information needs. This has placed limitations on the use of traditional databases as the sole providers of manufacturing information needs of the future.

Traditional Databases are not able to manipulate and link the different types of data. Object-oriented databases, however, provides facilities to handle a variety of data types, and would therefore be better suited to a manufacturing information environment as specified above. For the last three years development and implementation of such an MIS has been undertaken at Kenard Engineering in conjunction with Cranfield University, UK. The system has been developed from a solid foundation of proven existing systems. System modules have been developed where necessary and integrated within the overall MIS structure.

2. MODERN MANUFACTURING REQUIREMENTS

An understanding of the requirements needed in today's manufacturing environment is essential to the design of a MIS. Although a lot of different customer-manufacturer scenarios exist, an example of the manufacture of a prototype engineering component in a subcontract manufacturing situation will highlight many of the key requirements. Nowadays, such a component is more than likely to be designed by an engineer using a computer aided design (CAD) system. This component will almost inevitably be needed quickly so that the design can be qualified before pre-production [3].

The underlying requirement of an MIS is to provide salient information which enables each manufacturing stage to be carried out effectively and efficiently. This information, which can be gathered in the early stages of design and development, can be used to great benefit throughout the product life cycle. For example, in the design stage, a drawing file from a CAD system can be transmitted, via modem or other network connections, to the manufacturer, directly into a computer aided manufacture (CAM) station for value engineering, and obtaining manufacturing cycle times. The same information, adjusted if necessary after passing through the value engineering stage, can be used subsequently for the planning of the manufacture of the prototype part, and so on throughout the product life cycle. For the manufacture of complex components, stage manufacturing drawings can assist production. These can be produced from customers CAD files, by stripping out irrelevant details and leaving only the information required for that manufacturing stages. At the various stages along the product life cycle manufacturing methods will be refined and information will be updated accordingly, thereby making full use of information gathered during the development stages. By adapting this approach, valuable information evolves, whilst component development matures, therefore giving the full benefit of proven and reliable information collected and refined throughout the stages of manufacture. Benefits are realised for future production, when having pertinent and succinct information plays an important role in manufacturing efficiency and consistency.

The other key requirement for efficient prototype part manufacture is an effective way in which to present the user with relevant, job-related information. For example, in order to carry out a manufacturing task efficiently and effectively on the shopfloor, a machine tool operator needs to have access to a range of relevant information, such as finished component drawing, stage manufacturing drawings, detailed written instructions of the manufacturing sequence and, if the process has been pre-programmed, the part program together with the cutter paths and list of tools. The efficiency of the manufacturing process can be further enhanced by providing photographs of complex set-ups and special fixturing configurations. This information needs to be provided to the operator in a focused and accessible way, relevant to the job in hand and in a way that can readily maintained.

3. TASK-CENTRED DEVELOPMENT APPROACH

Traditional ways of providing manufacturing information to the shopfloor have not necessarily been task related. Rather general information was available but needed to be found or located as and when required. The prudent manufacturing organisations placed a strong emphasis on a thorough understanding of the task in hand, making sure that necessary information was readily available, and that the machine tool setter/operators should only be interested in information which would assist or is essential to the task in hand. With such diversity of tasks which need to be tackled within the manufacturing environment, confrontation with information which is not relevant to the task in hand is at best counter productive, and at worse could cause confusion.

The requirements of modern manufacturing described above highlight the need for a task-centred information environment. In contrast to the transitional IT solutions, in which tools are provided as individual applications and in which the user centres his activity around a selection of these mutually independent tools, with an object-oriented working environment such as Mirosoft Object Linking and

Embedding (MS OLE) and Apple's OpenDoc/Next, all the elements, necessary to support the user to carry out a specific task, can be brought together and presented as a single entity. This is an extremely useful concept as it concentrates on the management and organisation of data/tools in a way the user can understand, because it conforms to the way that humans do jobs every day. This concept seems to provide an ideal implementation foundation for the task-centred MIS framework, providing a general mechanism for MIS task/tool/data integration, so that the operator is given direct, structured and ready access to relevant information and tools [4,5].

4. SYSTEM DEVELOPMENT AND IMPLEMENTATION

A task-centred MIS has been developed and implemented within the MS OLE environment at Kenard Engineering, in conjunction with Cranfiled University, UK.

4.1 Company requirements

Kenard Engineering is a typical modern precision engineering machine shop, utilising computer numerically controlled (CNC) facilities and specialising in the making of aerospace and telecommunication components. It offers a service from prototypes through to, and including, production batches. Through an analysis of its particular needs and an extensive survey of currently available systems on the market, Kenard Engineering recognised the need for an MIS system that should be shopfloor based and could offer more than the standard direct numerically control (DNC) facilities by providing shopfloor information which are specific to the task in hand. This is needed to satisfy specific market demands so as to ensure a competitive edge.

In the quest to produce complex fully machined prototype components as quickly and efficiently as possible, an MIS named KIDS (Kenard Information and Data-collection System) has been developed around the needs of this company. However, the small-to-medium sized manufacturing companies have been kept very much in mind, because all manufacturing companies are faced with the conflicting dilemma of balancing the need to improve machine utilisation and operator efficiency, and the satisfaction of customer demands. The overall objectives of the system are:

- To set up a direct data link via modem, so that drawing files from customer's CAD system can be transmitted into the company's CAM system without the need to edit or reconstruct drawing elements.

- To allow the transmitted CAD drawing elements to be used to generate cutter paths ready for post-processing to any suitable and available CNC machine tool.

- To cut prototyping lead times, both by reducing CNC programming time and by reducing the time for CNC program verification at the prove out stage.

- To provide machine operators with job-related information in a focused and user-friendly manner.

4.2 System function

The information part of the KIDS system can be divided into two sections. The first part contains information which originates from the company's relational database including information relevant to customer's orders such as job cards and customer drawings. The second parts contains all other information such as other drawing files, CNC part program files and photographs, as shown in Figure 1.

The company's relational database had been developed in-house and contained information on orders and manufacturing methods for all manufactured components. The company operates a CAM system, capable of providing tool paths, and the generation CNC part programs. As the using of the customer's transmitted drawing files was developed, the information gathered were incorporated into the system as described above. At the development stage, it was felt important that if the system was to be successful, it had to be accepted by the user. Therefore, it was important that it should be easy to use and that any information required should readily accessible.

To display the relevant information, the first stage was to present the operators with a job card, which were already written and stored within the company database, so that they were identical to the familiar printed hard copies which already existed. The KIDS system incorporates data collection which monitors the manufacturing process, broken down between machine tool setting and cycle times of each operation. This data is collected via the job card menu and is activated by the operator clicking onto the time taken column for the relevant operation. The operator logging onto this operation is captured in real time and the event changes the colour of that operation the job card. This time logging information is collected and then used to calculate job cost, as well as being a management tool to appraise production efficiency.

4.3 System Structure and implementation

The main structure of the KIDS MIS has been written within the MS OLE environment using Visual Basic (VB). The system has been designed to retrieve structured numeric and character data from pre-defined fields within a traditional database, together with various other files, such as CNC part program files, CAD drawing image files and photographs. Figure 2 shows the overall structure of the system which has been implemented at the company. The rounded boxes denote shelf software packages, in some cases tailored to specified requirements. The system is entirely dependant on the existing company database. Feeding into KIDS is machine 'work to lists' for each machine tool which has been generated from a production scheduling package, drawing files (either received from customers or internally drawn), CNC part program files and photograph files. Shopfloor terminals which have been installed adjacent to each machine tool are connected by a local area network (LAN), as are the terminals within the production and planning offices.

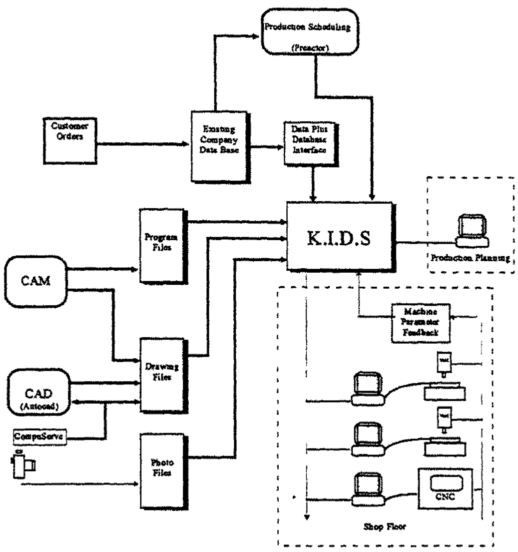


Fig. 1: KIDS System Structure.

The implementation of the KIDS system has been very much a step by step process, ensuring that each stage has worked successfully. To start, a DNC link was made from a single personal computer (PC) operated from a remote planning office and connected to the machine tools via RS232 ports and a switch box. Once it was clear that information other than CNC files could be displayed, PC's were introduce onto the shopfloor with up to three CNC machine tools connected to a single shopfloor PC. The first stage was to complete the DNC installation to each for the CNC machine tools. The next major change was to interface with the company relational database so that database information could be displayed within the information system

The component job card, taken form the database, acts as the menu for displaying information, as shown in Figure 2. In this instance a photograph of the actual component is displayed. Figure 3 shows the drawing of the same component which the operator can zoom into of better clarity if required. Through the use of VB it was possible design the job card front menu to assess other information requirements by clicking on the relevant part of the job card. This simple approach to information selection via the job card has been readily accepted by all users, and has allowed the system to evolve when information from other sources has been integrated. Figure 4 illustrates the task-centred way in which all the relevant information can be presented to the machine operator to effectively help his/her job on hand.

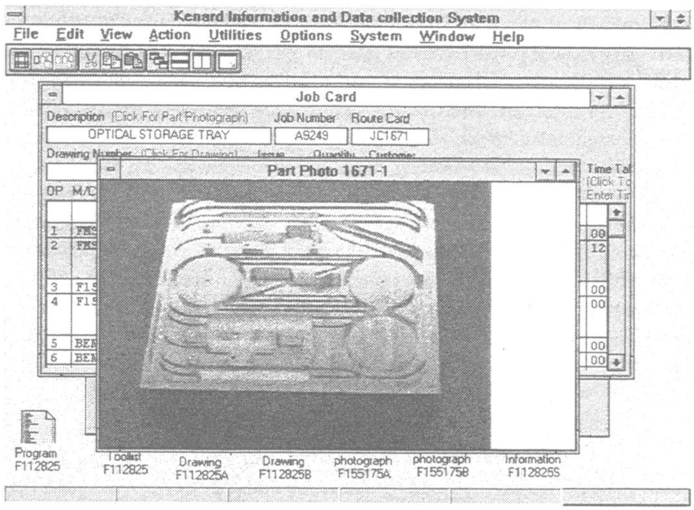


Fig. 2: Screen Shot of Manufactured Component displayed within KIDS

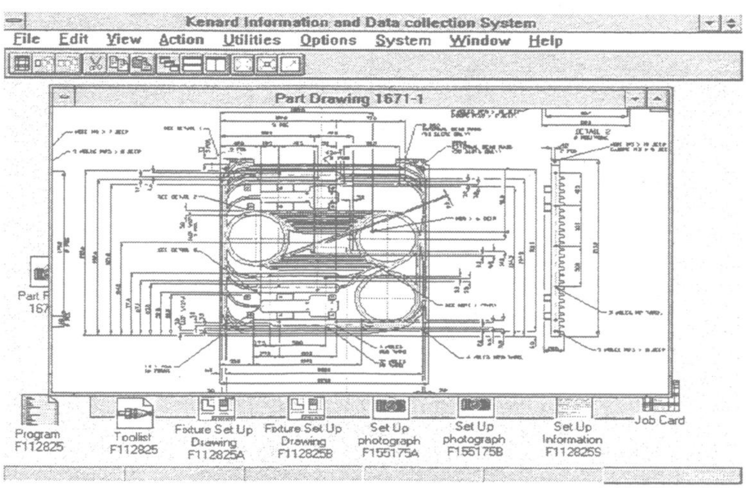


Fig. 3: Screen Shot of Customers Drawing displayed within KIDS

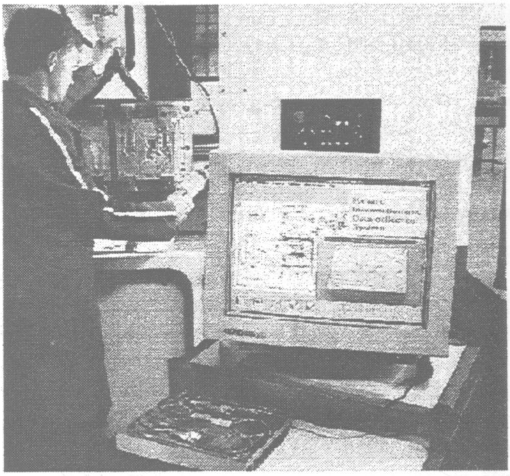


Fig. 4: Photograph of KIDS in operation

5. CONCLUSION

Kenard Engineering is a fairly typical small, but modern company. A task-centred MIS has been developed and implemented which is seen as an invaluable aid in this manufacturing environment. The KIDS system has successfully provided all relevant manufacturing information to the shopfloor using PC's on a local area network. This ease of access to information has allowed the operators to work more efficiently and has had the effect of reducing machine setting and operating times, while the improved quality of information has been reflected in product quality. In addition, because of the mutual goals shared, the company's customers also have a vested interest in making sure that drawing files were ready to use, so that manufacturing requirements such as drawing integrity and accuracy are guaranteed. Consequently, benefits of the system so far to the company include improved communication with clients, increased data accuracy, increased rapid prototyping ability, improved cost information, reduced set-up times and improved plant utilisation.

Although the system has been developed around the needs of this particular company, many other similar will be able to relate their common needs with the systems functionality and features. Due to its success, the KIDS system has been given the 1996 UK Machinery Award for Innovation in Production Engineering, for being "the most innovative application of computer technology in the manufacturing environment"

6. REFERENCES

1. Laudon, Information Systems A Problem Solving Approach, The Dryden Press, Florida, 1995.

2. Yeomans, R.W. Choudry, A and ten Hagen, P.J.W. Design Rules for a CIM System, , Elsevier Science Publishers B.V, Amsterdam, 1985.

3. Wu, B Manufacturing Systems design and Analysis, *2nd Ed,* Chapman & Hall,London,1994.

4. Wu, B. "The Open DSS Structure of Integrated Computer Aided Manufacturing Systems Design", The 14th International Congress on Cybernetics, August 1995, Namur, Belgium

5. Wu, B."An Overview of the Technical Requirements for an Integrated Computer-Aided Manufacturing Systems Design Environment", Int.J. of Manufacturing Systems Design, Vol.2, No.1, pp61-72, 1995.

MANAGING INTEGRATED MANUFACTURING THROUGH REAL-TIME MONITORING AND TQM

Henry, C. Tseng
Kingtronics Industrial Co. Ltd
Penthouse, Century Centre, 44-46, Hung To Road, Kwun Tong,
Kowloon, Hong Kong.

W.H. Ip
Department of Manufacturing Engineering
The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong

ABSTRACT

This paper addresses a methodology for the design and implementation of integrated manufacturing strategy in a labour intensive environment. The design encompasses the major innovations of manufacturing strategy which integrates Real-Time Monitoring System (RTMS) and Total Quality Management (TQM). The results show that it is effective and efficient to apply real-time monitoring system in a labour-intensive environment, and that TQM is the partner for the implementation to be successful. The integration strategy is considered to be a key to productivity improvement and an enhancement to competitive advantage. The architecture of the Real-Time Monitoring System and the information link of factory shop floor are also elaborated. Latest technology of hardware and software are also applied in the development life cycle of this project. This applied research project has been successfully implemented and its experience is valuable to the future development of manufacturing industry.

KEYWORDS

Manufacturing Integration, Integrated Manufacturing Strategy, Real-time Monitoring, TQM

1. INTRODUCTION

During the past ten years many manufacturing companies in Hong Kong have moved their production facilities into China to take the advantages of low labour and land costs. However, the experience indicated that the logistics of integration and communication of various functions such as marketing, production, material control etc. are often found to be ineffective due to the different management style and industrial environment of the production plants, and the overseas investors. Moreover, most of the Mainland workers are unskilled and do not have the required knowledge in production operations, hence the quality and yield rate are relatively poor when compared with other countries. Therefore, the company has investigated at an alternative strategy which will ensure their sustainable competitiveness. It follows a systematic manufacturing integration strategy and finally leads to the design and implementation of a Real-Time Monitoring system. It is found that the implementation of the RTMS forms one of the essential ingredient in the TQM culture.

The term real-time monitoring is referred to as in-process watching, supervising and inspecting some objects or process. The monitoring technique has found wide and successful applications in many fields such as mechanical, chemical, medical, aeronautical and astronautical fields etc. nowadays [1-3]. In modern production activities, the monitoring technology is usually used to monitor working states of machining processes and devices or assembly processes of the flexible assembly systems so as to ensure they work in health conditions. Researches into the real-time monitoring could be dated back to the early 1940s in the world. About in the 1960s, this technology began to be studied and utilized gradually in manufacturing processes with main objective of monitoring various abnormal cutting states by making use of sensors. In order to keep more advanced levels in the manufacturing field, many countries in the world are delivering considerable financial supports to foster, advance and speed up their manufacturing technologies. As one of the technological bottlenecks in advanced manufacturing technologies, the real-time monitoring technique is thereby strongly emphasized and developed quickly. Various published papers have been concerned with the subject of intelligent monitoring system [4-8]. In addition, the monitoring techniques have also been finding wide application in production management [9], underground coal-mine [10], steel production [11] and so on. Umscheid [12] suggested that using real-time monitoring system enhances SPC in a plastic moulding machine.

2. INTERRELATIONSHIP OF RTMS AND TQM

The application of RTMS in a plant producing electronic products which manual assembly dominated the assembly tasks may be a doubt to the industry. The fundamental reason underlying our approach is based on the doctrine of TQM, being the driving force of the company to achieve higher quality and productivity.

Total quality means that everyone should be involved in quality, at all levels and across all functions, ensuring that quality is achieved according to the requirements in everything they do. Total injects a systemic meaning of wholeness into quality. Every job is crucial and add to or detract from the quality endeavour. And lastly, by integrating the term management, the value of management responsibility is projected into the meaning of quality and wholeness already established. This responsibility is everyone's as total implies. Management responsibility does not necessarily refer to the company's Managers as it does in the entire quality literature that we have met. It can refer to the need for everyone to be responsible for managing their own jobs, which incorporates managers with workers and anyone else associated with the organization. The main principles of TQM can now be drawn out from the philosophy suggested by Flood [13], they are summarised as follow :

- There must be agreed requirements, for both internal and external customers.
- Customers' requirements must be met first time, every time.
- Quality improvement will reduce waste and total costs.
- There must be focus on the prevention of problems, rather than an acceptance to cope in a fire-fighting manner.
- Quality improvement can only result from planned management action.
- Every job must add value.
- Everybody must be involved from all levels and across all functions.
- There must be emphasis on measurement to help to assess and to meet requirements and objectives.
- A culture of continuous improvement must be established.
- An emphasis should be placed on promoting creativity.

The ten main principles provide a concise understanding of TQM as it stands today. In fact, TQM is no more than a kind of philosophy or thinking which is appropriate to a variety of management. Its essences may be comprehensively understood from different aspects and different point of views. For examples, the quality principles may be interpreted either through traditional management and organization theory or through viable system thinking as well as through socio-cultural systems thinking. The followings are the interpretations to TQM corresponding to the ten items as above from the viable system thinking and socio-cultural systems thinking, which encourage the company to develop real-time monitoring system even in a labour intensive environment.

❑ **Attitude of Willingness**

Viable • the intelligence function helps to discover what the client requirements are and sets the agreement about these within the quality mission; group and individuals agree customer requirements, and by taking responsibility for their jobs are able to work out effective ways of achieving them.

Socio-culture • all employees develop an attitude of willingness to provide a service to all internal and external customers, and hence seek to find out what their customers' needs are; jobs are more satisfying when we deal with happy rather than irate customers. This must be part of the rules and practices.

❑ **Commitment**

Viable • coordination and control at all levels aim to meet customer requirements, intelligence finds out how well the external quality goals are being met, and audit enables information about internal standards to be passed continuously to those who are responsible for controlling them (e.g., Quality Councils, groups and individuals).

Socio-culture • all employees have a commitment to ensure that the requirements of the customers they interact with are met; this is also part of the rules and practices.

❑ **Highlight Waste and Costs**

Viable • audit will highlight waste and costs; real-time control helps effective decisions to be made on up-to-date information, to prevent wasteful and costly management responses to information that can often be months old.

Socio-culture • those who have a commitment to quality objectives will seek out and wish to dispose of redundancy and wastage, and in this will be using their own time effectively and more meaningfully.

❑ **Preventative**

Viable • the management functions get deep inside the difficulty of reactionary policies; the future is planned to enable properly coordinated and controlled, informed and thought-out policies to enhance efforts to achieve quality.

Socio-culture • the rules and practices of the organization are preventative rather than reactionary.

❑ **Involvement**

Viable • management action comes from the five management functions and not from sovereign leaders according to traditional view; strategies are conceived and implemented by those who have close involvement with them; intelligence provides information to aid the planning process; all workers must plan and manage their jobs.

Socio-culture • a management plan for action is brought into being through quality management groups and by encouraging personnel to participated in and shape the management process.

❑ **Value Added**

Viable • added value is achieved because customer requirements are understood; jobs and tasks are organized to be effective in achieving customer requirements.

Socio-culture • this is another part of the rules and practices to be operated, so that all employees are working toward overall enhancement of quality in their work and personal quality of life.

❑ **Specific Goal**

Viable • the organizational scheme allows for involvement; goals are related to the whole and can be influenced from any management function or by any customer; quality methods like QCCs (Quality Control Circles) can be employed to encourage participation; supplier development is an innovative initiative in quality management that emphasizes the breadth of meaning of this principle.

Socio-culture • the rules and practices of the quality culture pervade every corner of the organization, which requires a cohesive underlying constitutive meaning that makes them meaningful to all employees; human needs are satisfied; everyone has a share in the quality concept.

❑ **Performance Indices**

Viable • a comprehensive set of performance indices that get to the heart of what is happening in an organization are recommended and complement the five management functions.

Socio-culture • this is another part of the rules and practices that should be normalized, relating to both work and quality of life.

❑ **Learning**

Viable • the cybernetic principles give rise to inherent continuous change encouraging progressive quality improvement; the organization learns through intelligence, groups and individuals learn through participation, learning provides the basis for continuous improvement.

Socio-culture • each person when doing or redesigning their job, must achieve some kind of measure that tells them whether their quality objectives are being met; this is another part of the rules and practices.

❑ **Creativity**

Viable • with intelligence and leaning functions in place throughout the recursive organization, and with plenty of relevant real-time information, an organization is prepared for and can encourage creativity.

Socio-culture • all people should participate, each understand better than most their own job, each holds different perceptions about other's job, the department or organization as a whole; their bringing together in participatory groups will thus release much otherwise stifled creativity; in many ways the most valuable resource an organization has is its human element, not its technology.

It is obvious that those interpretations of TQM from either "viable system thinking" or "socio-cultural system thinking" viewpoint as stated above indicate that TQM requires the RTMS as the driving agent, their interrelationships are further elaborated into the following:

- *real-time control* helps effective decisions to be made on up-to-date information; those who have a commitment to quality objectives will seek out and wish to dispose of redundancy (*Highlight Waste and Costs*).
- the future is planned to enable properly *coordinated* and *controlled* (*Preventive*).
- intelligence provides *information* to aid the planning process (*Involvement*).
- *jobs and tasks are organized to be effective* in achieving customer requirements (*Value Added*).
- *everyone has a share in the quality* concept (*Specific Goal*).
- *learning* provides the basis for continuous improvement (*Learning*).
- with intelligence and leaning functions in place throughout the recursive organization, and *with plenty of relevant real-time information*, an organization is prepared for and can encourage creativity (*Creativity*).

The representation of interrelationships is shown on figure 1.

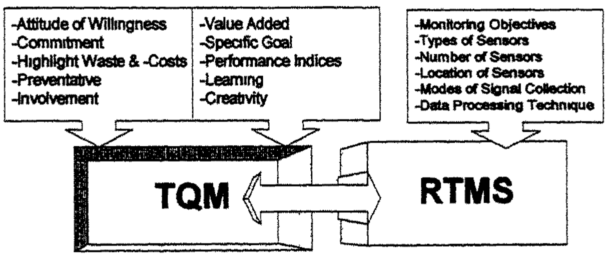


Fig. 1: Interrelationships of RTMS and TQM

3. THE IMPLEMENTATION

The implementation of RTMS requires two major elements; reliable sensors and data processing technology. Therefore, the following concerns will be faced in order to develop the real-time monitoring system:

1. *Monitoring objectives*
   First of all, it must make clear that what the company exactly wants to monitor and the goals the company would like to achieve, because this relates to which kinds or sensors should be chosen in the real-time monitoring system to be developed;
2. *Choosing types of sensors*
3. *Number of sensors*
   This will affect how to select interface that is capable of accommodating all signals from sensors and ensure collecting data in real-time without data lost;
4. *Location consideration of sensors' installation*
5. *Consideration on working modes of signal collection*
   Because there will be more than one hundred signal sources from work stations and other kinds of signals received from key-in devices that may take place at the same time, the computer used in the monitoring system would be required to have the capability of handling lost of events that have already happened, so a compromise strategy namely the appropriate working mode is needed to be made up in order guarantee collecting data quickly and precisely.
6. *Data processing technique*

3.1 Real-Time Data Collection System Based

In the implementation, there are over hundred work stations as well as several manual quality test stations (QC checkers) intermixed among them in an assembly line. Therefore, the selection of appropriate system hardware and software is considered to be important. The worker performs assembly task assigned to him or her in the work station. Final quality test of the products is carried out automatically by an automatic computer controlled test station. The evaluation of the workers' performance depends on how much time he or she spends on assembling one unit and the amount of units that has been assembled. As for the quality of assembled products, it will not be known until a manual test station has completed the check. The tester will supply with testing report on product quality. It is obvious that at least two kinds of signals would be measured; one is the number of assembled units and the time they spend in assembly; the second is to detect the existence of objects and counters to count the number of assembled units of each work station, and at the same time, noting down assembly time the worker has spent. These can be achieved through a reporting system which is elaborated as follows.

3.1.1 Report information input methods from QC test station

Like assembly work station, the report information from QC test stations must also be sent to the host computer in real-time. However, the contents of the report are much more rich and complicated than the photoelectric switch signal. Dedicated key-in devices and bar code readers can be chosen for the data capture scheme. They are developed by the company and considered to be an economical method for sending report information. These devices use parallel data bus to connect with the host computer through a special buffer interface. The developed devices are depicted in figure 2.

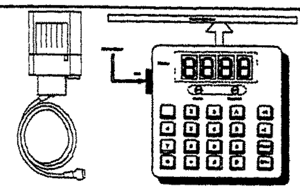


Fig. 2 : A dedicated key-in device and a bar code reader

In this device, a single chip CPU is used to control all works. Operator can input error codes with timely stamped. Two lamps are designed to indicate "ready" to send and "received" already so as to establish inquiry and acknowledgement between the device and the host computer. Another function of the device is to detect signal automatically from the photoelectric switch sensor. All QC test stations and work stations use the same input devices with the exception that there is no interface designed for sensor signal input in the device of the QC test station. The operator in the work station can input assembly information manually through the device. The hardware configuration of the system is presented in figure 3.

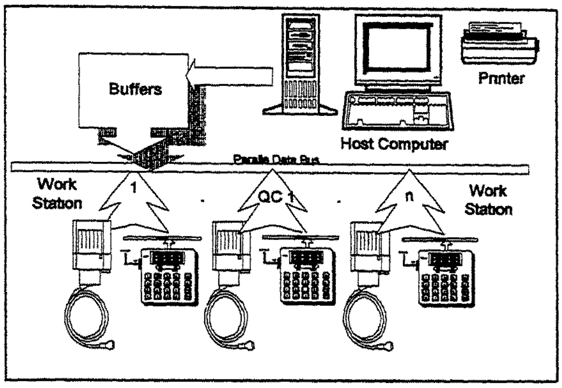


Fig. 3 : Hardware configuration of the monitoring system

4. CONCLUSION

The implementation of TQM and RTMS in the manufacturing company is the result of an integrative approach, where the production planning, the shop floor management and quality control are linked up. This is a integrative management model that has never appeared in a labour-intensive manufacturing environment. Today, this approach has not been experienced in China or other Far East countries. The implication of employing monitoring system is much more greater beyond the fact itself, it may affect culture and the consciousness of product quality in the people's minds, so that stimulating everybody working in the enterprise to improve their work quality, and further products quality will be improved. After the implementation, the ingredients become one portion of the basic diathesis of the enterprise. From another perspective, the RTMS has also become a kind of catalyst to promote enterprise TQM culture. The reason to enable this change is obvious, because the monitoring system makes it possible for everyone in the factory to be involved to meet the needs of TQM.

5. REFERENCES

1. Fu, Jachih J. C., "Tool Wear Monitoring in Face Milling by Integration of Static and Dynamic Signals Using Neural Networks", Ph. D. Dissertation, Lehigh University, 1992.

2. Liu, T. I., Ko, E. J., "On-Line Recognition of Drill Wear Via Artificial Neural Networks", Proceedings of Winter Annual Meeting of the American Society of Mechanical Engineers, Dallas, USA, 1990.
3. Lau, Hui Kong, "An Expert System to Monitor Drill Condition", Ph. D. Dissertation, University of Windsor (Canada), 1992.
4. Camarinha-Matos, L.M.; Lopes, L. Seabra; Barata, J., "Execution Monitoring In Assembly With Learning Capabilities", Proceedings of the 1994 IEEE International Conference on Robotics and Automation, San Diego, CA, USA, 1994.
5. Syed, A.; ElMaraghy, H.A.; Chagneux, N., "Real-Time Monitoring And Diagnosing Of Robotics Assembly With Self-Organizing Neural Maps", Proceedings of the IEEE International Conference on Robotics and Automation, Atlanta, GA, USA, 1993.
6. Milberg, J.; Wisbacher, J., "Acoustic Test Procedures. A Powerful Method For Quality Assurance And Process Monitoring In Assembly", Annals of the CIRP, Vol. 41/1/1992.
7. Basanez, L.; Torras, C.; Ilari, J.; Sanfeliu, A., "Operation Specialists For Automatic Programming And Monitoring Of Robotics Assembly Cells", Robotics and Computer-Integrated Manufacturing, Vol. 6 No. 4, 1989.
8. Lopez-Mellado, Ernesto; Alami, Rachid, "Execution Monitoring System For A Flexible Assembly Workcell", Proceedings - 16th International Symposium on Industrial Robots; 8th International Conference on Industrial Robot Technology, Bruassels, Belg, Sep 30-Oct 2, 1986.
9. Charalambopoulos, A.; Vathis, J., "On-Line Production Monitoring System", Proceedings of the IFIP TC5/WG5.7 5th International Conference on Advances in Production Management Systems - APMS'93, Athens, Greece, 1993.
10. Raman, Ravi S.; Nutter, Roy S. Jr.; Reddy, Y.V. Jr., "Production System For Intelligent Monitoring Systems:, IEEE Transactions on Industry Applications, Vol.24, No.5, 1988.
11. Coudal, Edgar F., "Process-Monitoring Raises Production, Improves Quality", Manufacturing Systems, Vol.6 No.9, Sep. 1988.
12. Umscheid, Tomas E., "Production Monitoring Enhances SPC", Quality Progress, Vol.24, No.12, Dec. 1991.
13. Flood, Robert L., Beyond TQM, John Wiley & Sons, New York, USA, 1993.

THE DEFINITION OF A MANUFACTURING STRATEGY ANALYSIS/MANUFACTURING SYSTEMS DESIGN INTERFACE

R.S. Hull and B. WU
School of Industrial and Manufacturing Science
Cranfield University
United Kingdom

ABSTRACT

If technologies such as automation, CAD or CIM, or new practices such as concurrent engineering, JIT, TQM, etc., are to be successfully introduced, then a strategic foundation is often deemed necessary. The strategy needs to be effectively communicated to the members of the organisation and understood by those responsible for implementing the change. It can then be transformed into action plans and manufacturing systems design projects with clearly defined objectives and detailed task definitions. However, companies often encounter problems in implementing strategic vision, particularly in the creative process of defining action plans. One reason is the lack of tools to assist them in this process. The linking of a task-centred methodology to action plans is one means of providing a case specific approach. This paper reports the current state of development of a module linking manufacturing strategy analysis techniques to a task-centred manufacturing systems design methodology. It proposes a semi-structured, computer-supported approach which forms part of an integrated and open computer aided manufacturing systems design environment.

KEYWORDS

Manufacturing Strategy Analysis, Manufacturing Systems Design

1. INTRODUCTION

The re-organisation of manufacturing facilities and the introduction of new technologies and techniques can be greatly assisted through the application of structured approaches and methodologies. However, it is generally accepted that structured approaches are not adequate in themselves to guarantee the successful implementation of the new concepts. Key issues, which often contribute towards the failures of advanced manufacturing technologies and flexible manufacturing installations or the unsuccessful adoption of new manufacturing philosophies, include a lack of a strategic link between the technology and the business objectives [1] and the lack of strategic initiatives with respect to the implementation of technological and organisational change. Indeed, it has been suggested [2] that trying to improve manufacturing by simply adopting new techniques such as JIT, TQM, etc., is no longer an effective strategy, since what is now required is a strategy which specifies the kind of competitive advantage that the organisation is seeking through the manufacturing function.

Hence, when a manufacturing systems design (MSD) project is initiated, an important stage in the process should be the execution of a manufacturing strategy analysis (MSA). This should aim to assist the MSD project engineers to capture and understand their organisation's manufacturing strategy and to use the relevant parts of the strategy to aid them in the systems design process. Once the strategy capture and analysis has been achieved then the design team can commence with the execution of their MSD project. To help structure the transfer from strategy analysis to systems design and to manage the MSA/MSD interface, this paper introduces a three stage approach, which applies three frameworks: a strategic framework, an action plan framework and a systems design framework. The paper is divided into four parts. Section two describes the problem domain and outlines why a MSA/MSD interface is required. Section three discusses the concepts behind the proposed module and section four describes the typical procedure, applying the MSA/MSD concepts within a generic task-centred framework to construct a specific design approach. Finally, applications and future research directions are discussed.

2. THE PROBLEM DOMAIN

It has been observed that there have been few attempts to *'translate'* manufacturing strategy into decision processes for manufacturing system design [3]. There have also been few attempts to link strategy to MSD projects and the methodology to be adopted. Typical approaches to the linking of strategy to projects or action plans include Ward *et al* [4], who attempted to map manufacturing concerns to action plans. The approach adopted was that of a factor analysis based on empirical case study evidence. It related a number of individual manufacturing concerns to ten specific factors, which in turn were compared to the typical strategic decision categories. In addition, a further set of concerns were related to eleven programme factors to provide an indication of planned action plans aimed at improving the business and manufacturing performance. More recently, Kim and Arnold [5] have introduced a framework linking competitive priorities, manufacturing objectives and actions plans. Such approaches do not enter the subsequent phase of systems design methodology, and a gap between the content and formulation of manufacturing strategy and the design of manufacturing systems, with respect to the methodologies utilised, can be observed.

A second issue concerns the strategy analysis and systems design interface itself. The research and practice fields of both manufacturing strategy and manufacturing systems design have tended to develop independently of each other as well as independently of the related management and business research [6]. The result is that a somewhat ill-defined interface has been produced. Figure 1 illustrates a typical example of the problem. The definition of the MSA/MSD boundary is influenced by the extent to which strategy is considered to provide a sense of direction for the enterprise and the manufacturing function, and the extent to which it is providing a co-ordinated set of decisions defining the role of the manufacturing function and the deployment of its resources. In addition the operationalisation or implementation of manufacturing strategy has been found to be a difficult task. The common feature of many formulation and implementation approaches [7-12] and those implied within the strategy process and content literature [13-15], is that of the development of action plans of increasing detail and commitment. The translation from a set of strategic decisions to a set of action plans is not a clearly specified procedure. In many case it relies on the experience and knowledge of senior management. It is often viewed as a creative process requiring a large degree of imaginative and holistic thinking [10,16].

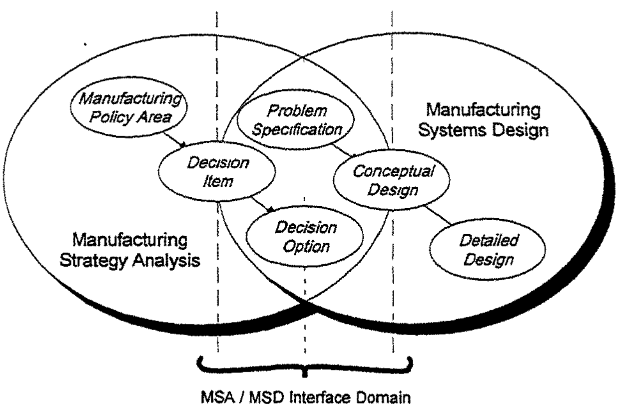


Fig. 1 : The MSA/MSD interface

This is reinforced by Greenhalgh [7], who observes that failure to achieve objectives can be averted if the activities required to attain those objectives are rigorously considered and hence, that if the action plans based on strategic objectives have received considerable thought in their specification. The detailing of such action plans is one of the most poorly executed elements of strategic business and operating planning [10], to the extent that they are often not specified in adequate detail to aid implementation. Hence, a structured approach to action plan detailing and translation would assist manufacturing strategy implementation and MSD project definition.

3. INTERFACE PRINCIPLES AND CONCEPTS

As part of the research into an integrated and open computer aided manufacturing systems design (I/O-CAMSD) decision support framework, manufacturing strategy is being used as a principle input for the MSD process. The MSA/MSD interface is to serve two basic functions. Firstly, to provide the relevant contents of the manufacturing strategy with which to define the MSD Project Terms of Reference. Secondly, to provide an indication of the focus and direction of the MSD project, with respect to the methodology to be followed, and an indication of the relevant strategic contents that have an effect on the systems design. In order to provide this functionality and structure the translation from strategic decisions to systems design, the MSA/MSD interface applies three fundamental frameworks, as illustrated in figure 2. These comprise a generic framework for manufacturing strategy, a framework for action plans and the I/O-CAMSD framework for systems design. Within and between each framework a series of links and tables relate the constituent concepts, decisions, plans and design tasks.

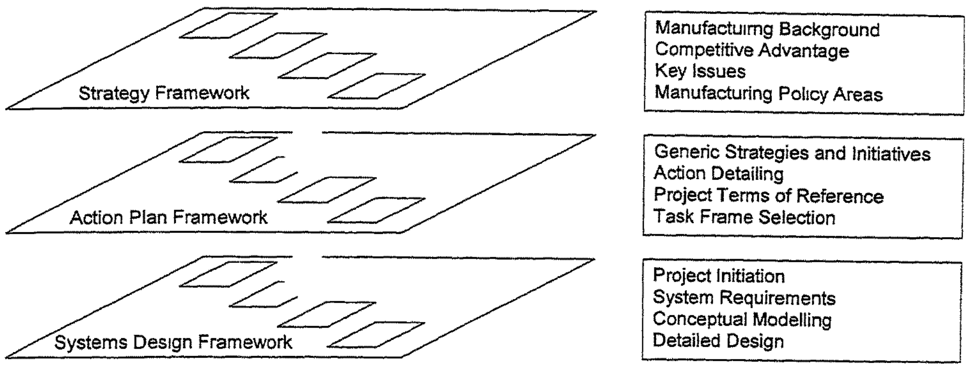


Fig. 2 : The MSA/MSD interface frameworks

3.1 The Strategy Framework

The manufacturing strategy framework is based on an analysis of the literature with respect to the content and process of manufacturing strategies. The framework is based on the format of a manufacturing strategy document comprising four key elements: background and the role of the manufacturing function; statement of competitive advantage, statement of the key issues affecting manufacturing; and development of the strategic aims. A manufacturing policy area model forms a major feature of the last section of the framework. This policy area model can be considered to be a derivative of the models developed by Hayes et al [14,15]. It is composed of eleven separate policy areas: capacity, facilities, processes and technologies, vertical integration, supplier relations, human resources, quality systems, production planning and control, product scope and new products, performance measurement, and organisation. Each policy area has been defined with respect to its decisions, sub-decisions, options, parameters and influences.

3.2 The Action Plan Framework

The action plan framework is composed of a series of generic operating and action plans. Each of the generic action plans are associated with a set of MSD tasks derived from the MSD framework. Other elements associated with the action plan framework include the project initiation stage of the I/O-CAMSD framework, in particular the manufacturing criteria and the project terms of reference consolidation. The manufacturing criteria relate the manufacturing strategy to the manufacturing system by requesting the project team to define the system purpose, the system performance, the system characteristics and the system cost structure. The project terms of reference consolidate the information obtained from the manufacturing strategy, the manufacturing criteria and an analysis of the existing manufacturing system in order to define the particular MSD project being undertaken. In addition, the terms of reference provide a means of deriving the evaluation criteria for the MSD project.

3.3 The Systems Design Framework

The systems design framework is composed of a series of task frames, with each task frame containing a series of design tasks related to particular manufacturing subsystems at specific stages in the design problem solving cycle [17]. The design tasks within this framework have been derived from an extensive analysis of available in the literature and industrial practice. The I/O-CAMSD framework and the task centred methodology, which constitute the design framework, have previously been presented [17-19].

3.4 Framework Relationships

A simplistic mapping of the eleven strategic policy areas onto the three principle sub-systems provides the first stage of task frame guidance, in terms of indicating the primary area of focus of the project. A second stage links the primary, secondary and tertiary relationships between the policy areas and the individual task frames for each level in the systems design framework. A third stage associates decisions and decision options to individual design tasks. In order to assist in the customisation of a generic task list, a fourth relationship table is required, within which the project terms of reference are linked to the task frames and the design tasks.

4. PROCEDURE

The first stage of the MSA/MSD interface is to understand and capture the manufacturing strategy. In order to facilitate this procedure and provide the MSD team with the rationale behind the strategy, it is suggested that the strategy capture process should follow the format of a strategy formulation process as a means of constructing a strategy document. This also has the added advantage of providing guidelines for the formulation of a manufacturing strategy if one does not already exist. Figure 3 presents a high level view of the process for the first stage, indicating the individual steps and their contribution to the strategy document. It is largely based on manufacturing audit approaches, as described in the literature [8,11]. However, the application also incorporates the benefits which a computer aided framework can provide [20], with the addition of simple tools and techniques to assist the strategy capture and development, particularly at the latter stages in the process.

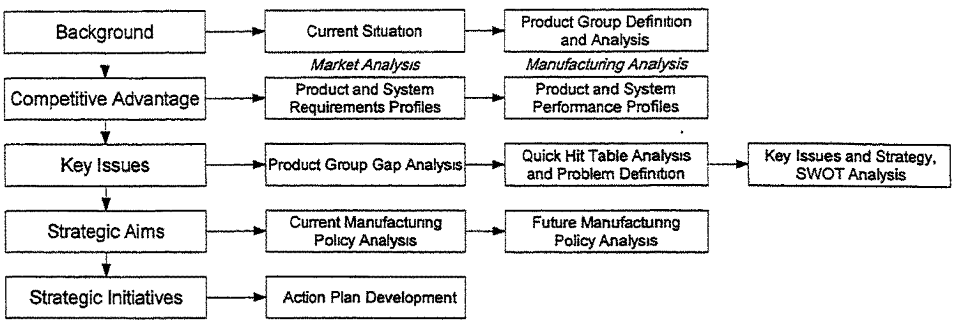


Fig. 3 : Strategy formulation and capture

In addition, through the storage and re-use of data, decisions and recorded assumptions, the strategic information and rationale behind the strategies can be retrieved swiftly throughout the strategy formulation and MSD processes. As well as simple data capture methods, additional tools which have been developed include graphical analysis, relationship tables and associated on-line help facilities. An example within the interface module is the 'quick hit' table, which provides an indication of the typical problems prevalent in each of the manufacturing policy areas and their effect on the competitiveness of the manufacturing systems, with respect to six key competitive criteria, and vice versa.

The second stage of the interface, often considered as part of the strategy itself, is the derivation of action plans. For this purpose a series of generic strategies are provided. These are prioritised, customised, expanded and augmented by the design team in order to suit the specific needs of the business and to align its strategy to the various markets. Where several action plans have been specified, each will require detailing. From an understanding of the contents of these action plans, derived during their detailing, they can be grouped into MSD projects. This provides the designers with a simple mechanism to ensure that the MSD project to be undertaken is compatible with the manufacturing strategy. Once the rationale behind the strategy and the strategy itself has been captured and understood, the next stage is to define the MSD project and derive the project terms of reference, the objectives, scope and constraints of the project. This information, in conjunction with the strategy, provides the scope, focus and direction of the project and assists in the selection and customisation of a suitable project methodology. Hence, a single MSD methodology is not provided. In order to remain open and flexible, a task-centred approach is adopted which is structured according to systems theory and the basic problem solving cycle. Since each of the generic action plans is associated to a particular set of task frames and design tasks, a design process plan or a route through the systems design framework can be constructed.

Thus the final aim of the interface is to guide the users through their MSD project, suggesting the appropriate sequence of task frames and related design tasks to execute. The first step in achieving this is the selection of task frames, based on the tables of the framework relationships previously described. The second step concerns the selection of the appropriate design task, again using the framework relationship tables for guidance. The design task is then configured to meet the specific requirements of the systems engineer, before it is finally executed. The collection of design tasks within a particular task frame contributes towards the development of a concept. The aggregation and consolidation of these concepts provides the systems design.

5. APPLICATIONS AND FURTHER WORK

The I/O-CAMSD decision support framework aims to assist manufacturing engineers design, or redesign, their manufacturing system. Whilst it would seem reasonable to assume that the stimulus for such a project would have involved the development or review of the organisation's manufacturing strategy, this will not always be the case. A recent strategy development exercise carried out in a company operating in the aerospace industry revealed that whilst the organisation possessed a willingness to improve operations, it still lacked the willingness to investigate what it needed to improve and by how much it needed to improve. This was further exacerbated by the lack of an existing manufacturing strategy and the view at the director level that manufacturing's role was simply to empty the order book, without any consideration of whether it could contribute towards providing a competitive advantage for the company.

The strategy capture stage therefore provides an additional function, that of allowing the MSD project team to review their manufacturing strategy in order to ensure that the aims and objectives of their intended project do in fact agree with the strategy. Alternatively it allows them to question the objectives of their MSD project and the premise of undertaking the project without a manufacturing strategy or without a clear understanding of their strategy. The I/O-CAMSD framework and MSA/MSD interface merely provide generic guidance for action plan detailing and design task selection. The important input is that of the members of the project team who must customise the suggested MSD methodology to suit their specific case. The interface is therefore contributing to solving the common problem of not knowing where nor how to commence.

Following discussions with industrial collaborators, further investigations into the MSA/MSD interface have been identified. These derive from, and relate to, the need to provide a more detailed picture of the current situation, particularly with respect to including techniques for identifying how the organisation, and the manufacturing function, has reached its state. 'Technology Road-Mapping' has been suggested as a possible approach. It was also suggested that the provision of weightings to the linkages within the framework relationship tables could possibly prove useful.

6. CONCLUSIONS

This paper has introduced the underlying concepts of a MSA/MSD interface which serve to specify the terms of reference of a MSD project and to assist in planning the design process. The principle components of such an interface have been defined with respect to a structured approach to the capture and, where necessary, development of a manufacturing strategy, a structured approach to the development of action plans with which to implement the manufacturing strategy and a structured and systematic means of deriving the terms of reference of the MSD project from the manufacturing strategy. Once the project has been fully specified, and its strategic foundation is assured then the I/O-CAMSD framework and related task frames and design tasks can be associated to particular manufacturing policy areas, strategic decisions and strategic options. Through the incorporation of the manufacturing strategy into the planning of the design process, customer and market driven issues are addressed resulting in a greater chance that the resultant manufacturing system will be able to support the corporate objectives and provide a competitive advantage.

7. ACKNOWLEDGEMENTS

The work reported here was executed under EPSRC UK Grant ref. GR/J 84656. The authors would also like to thank Edwards High Vacuum International Ltd., Information Engineering Ltd., Kenard Engineering Co. Ltd., LucasVarity and Synchronized Manufacturing Ltd. for their support.

8. REFERENCES

1. Hayes, R.H. and Jaikumar, R., 'Manufacturing's crisis: new technologies, obsolete organisations', Harvard Business Review, 66, September/October, pp 77-85, 1988.
2. Hayes, R.H. and Pisano, G.P., 'Beyond World-Class: The New Manufacturing Strategy', Harvard Business Review, 72, January/February, pp 77-86, 1994.
3. Bennett, D. and Forrester, P., Market-Focused Production Systems. Design and Implementation, Chapter 6, Prentice-Hall, London, 1993.
4. Ward, P.T., Miller, J.G. and Vollmann, T.E., 'Mapping Manufacturing Concerns and Action Plans', International Journal of Operations and Production Management, 8, 6, pp 5-18, 1988.
5. Kim, J.S. and Arnold, P., 'Operationalizing manufacturing strategy. An exploratory study of constructs and linkage', International Journal of Operations and Production Management, 16, 12, pp 45-73, 1996.
6. Leong, G.K., Snyder, D.L. and Ward, P.T., 'Research in the process and content of manufacturing strategy', OMEGA, 18, 2, pp 109-122, 1990.
7. Greenhalgh, G.R., Manufacturing Strategy, Formulation and Implementation, Addison-Wesley, Sydney, 1990.
8. Hax, A.C. and Majluf, N.S., The strategy concept and process, a pragmatic approach, Prentice Hall, London, 1991.
9. Hill, T., Manufacturing Strategy, Text and Cases, Chapters 2 & 4 , Macmillan, Basingstoke, 1995.
10. Judson, A.S., Making Strategy Happen, 2nd Ed., Blackwell, Oxford, 1996.
11. Platts, K. and Gregory, M., Competitive Manufacturing, A Practical Approach to the Development of a Manufacturing Strategy, IFS Publishing, Kempston, 1988.
12. Samson, D., Manufacturing and Operations Strategy, Prentice Hall, Sydney, 1991.
13. Skinner, W., Manufacturing: The Formidable Competitive Weapon, John Wiley, New York, 1985.
14. Hayes, R.H. and Wheelwright, S.C., Restoring Our Competitive Edge, John Wiley, New York, 1984.
15. Hayes, R.H., Wheelwright, S.C. and Clark, K.B., Dynamic manufacturing, creating the learning organization, Free Press, New York, 1988.
16. Riis, J.O., 'Integration and manufacturing strategy', Computers In Industry, 19, pp 37-50, 1992.
17. Hull, R.S. and Wu, B., 'Computer Aided Manufacturing Systems Design An Open and Flexible Design Methodology', Advances In Manufacturing Technology X, Proceedings of The Twelfth National Conference On Manufacturing Research, Bramley, A.N., Mileham, A.R., Owen, G.W., pp 126-130, University of Bath, Bath, 1996.
18. Wu, B., 'An overview of the technical requirements for an integrated computer-aided manufacturing systems design environment', International Journal of Manufacturing Systems Design, Vol. 2, No. 1, pp 61-72, 1995.
19. Wu, B. and Hull, R.S., 'A Task-centred Methodology to Support an Integrated and Open Computer Aided Manufacturing Systems Design Environment', Proceedings of the International Conference on Manufacturing Automation, University of Hong Kong, Hong Kong, 1997.
20. Wu, B., 'Computer-aided manufacturing systems design: framework and tools', Control and Dynamic Systems, Advances in Theory and Applications, C.T. Leondes, Vol. 47, pt 3, pp 281-324, Academic Press, 1991.

Object-Oriented Modelling and Development of Flexible Manufacturing Cells

L. K. Chu & C.K.F. Chiu
Department of Industrial and Manufacturing Systems Engineering
The University of Hong Kong
Hong Kong
(E-Mail: lkchu@hkuxa.hlu.hk)

Abstract: This paper describes an object-oriented approach to the development of control software for flexible manufacturing systems (FMS). The design of the software is performed within a new framework of FMS design, implementation and operation. The implementation of the proposed software is highlighted by a generic flexible manufacturing cell (FMC) control software architecture and an integrated development environment for FMC's. The control software architecture consists of a three-layer hierarchy including the operation, networking and operating-system layer. This arrangement succeeds in separating the hardware and communication issues from the main FMC control applications. On the other hand, the proposed development environment integrates several development tools to support system development from design, evaluation, implementation, testing to deployment. Such a development environment streamlines FMS development process and saves both time and achieves better FMS design.

Keywords: flexible manufacturing cell, flexible manufacturing system, integrated development environment (IDE), intelligent shop-floor device (ISFD), object oriented programming (O-OP).

1. INTRODUCTION

It is well argued and well documented that FMS takes a long time to design and implement. The behaviour and structure of FMS's epitomise a complex system that their design and implementation call for expert design knowledge and techniques, and development tools especially for control software development. It is hard to strike the optimum configuration without the aid of design tools, e.g. mathematical models for economic justification, group technologies algorithms for forming part-machine families, queuing network and simulation for system modelling and design. On the other hand, the major obstacle to implementation lies in the generation of FMS control software. Apparently, the planning and control functions of a reasonably sized FMS are legion. To implement so many software modules is already a difficult task, to achieve the levels of sophistication as suggested by the research community is beyond the practitioners' capability. Poor system design and implementation of control software of the earlier installed systems caused many problems. They include low production rate and inability to respond to product and volume changes. In other cases it is found that such systems are difficult to maintain and to modify or upgrade.

To tackle the above problems, this paper suggests several new viewpoints. An FMS is considered as a dynamic and complex configuration of software and hardware entities. Due to the need for maintenance and continuous upgrade, it evolves during its lifetime. The best strategy to operate an FMS, therefore, is to let the user itself perform such maintenance and upgrading processes, with minimal resort to the supplier. To attain this goal, the company's involvement in the system development process (i.e. design and implementation) must be substantial. Unfortunately, this ideal is probably not a common practice to date. The major difficulty arises from the fact that the development of FMS is a complicated and highly specialised job. Also, the required development tools are not always available easily. For such typical one-off project, it is not justifiable, in terms of time and cost, to keep an in-house project team to perform the development from scratch. The solutions to this problem consist of the availability of such development tools. Many FMS modelling and analysis tools have been available though it is desirable to put them in a unified framework. The tools for the development of FMS control software, however, have not been dealt with effectively. A major reason is that individual FMS designs are specific. Further, to facilitate system evolution, some form of operating environment (essential software) is needed by the FMS user.

Another useful viewpoint is in locating FMS control software problem in the web of FMS development and operation processes. The solution to the software problem cannot be obtained by solely tackling the problem *per se*. A holistic approach with the consideration of the various aspects of FMS is necessary. Moreover, as advocated in this paper, the FMS control software should fit in the design-implementation-operation cycle. As mentioned above, modelling and design of FMS requires putting these tools in a unified framework. To continue this argument, the generation of control software follows from the design stage and their relationship is due to the propagation of the design configuration from design to implementation. Again, an integrated environment for design and software generation is called for.

To address the FMS development problem, a generic FMC control architecture as a reference model for software development is established. Additionally, a set of tools are provided under an integrated development environment (IDE) to facilitate the design and implementation of FMC's. The control software has a generic structure which can be adopted to suit a wide range of system requirements. The control functions provided are comprehensive and the control software can be easily maintained and modified. Central to the IDE comes the concept of object oriented programming (O-OP) and software reusability. O-OP is useful for modelling most FMS entities and their complex relationships. The task of generating FMS control software based on the O-O approach will be facilitated. Also, O-O structured approach to software design has been well developed. This will not only encourage rigorous software engineering effort but also the use of reusable software modules. This paper describes a prototypical system called FMC-IDE (IDE for flexible manufacturing cells) which embodies the above viewpoints.

2. A REVIEW ON ISSUES RELATING TO FMS CONTROL SOFTWARE

There are several significant issues which are critical to successful FMS development. These include the availability of suitable system architecture; reusable software; and the use of appropriate system design tools.

2.1 Open system architecture

Open FMS development can be facilitated based on an open-system reference model. An open system may be defined as exhibiting the characteristics of high connectivity between subsystems, availability to a broad set of users, expandability of both hardware and software, and portability of software from one subsystem to another. Wright[1] adopts an anthropomorphic paradigm to derive the skeleton of the whole CIM operation architecture. He suggests the use of *de facto* standards for languages, operating systems, and computer hardware to serve as a basis for open system development, which results in higher feasibility of use and in turn acceptability. Levin et al[2] suggests a centralised control software structure with emphasis on software modularity, and accounting of real-time requirements in programming. The National Bureau of Standards (NBS) proposed the concept of virtual manufacturing cells and automated manufacturing research facility[3]. The system is optimised dynamically by integrating planning and scheduling functions together.

2.2 Software reuse methodologies

Nof[4] identifies the advantages and limitations of using object-oriented methodology in manufacturing. He notices that the natural and intuitive correspondence, inherent structure, modularity, discipline, adaptability and flexibility are beneficial to manufacturing applications. However the complexity in handling distributed communications and operations, the difficulties in co-ordinating remote activities, the potential high run-time cost, and a lack of functional support and standard libraries limit its use. The reusable software concept is also studied by Smith[5]. Important reusable software concept including the use of scaleable architecture, automatic generation of control code and object-oriented design are identified. They are achieved through the use of hierarchical control skeleton, context free grammar modelling approach and object-oriented modelling of physical entities, respectively.

2.3 System design methodologies

Talavage[6] identifies variables which affect system performance. These variables include process plans, materials, configuration and operation control logic. By using simulation, knowledge-based system technique and problem decomposition method, an optimal solution as a combination of the aforesaid variables is found automatically. The problem decomposition method relates practical problems to such variables. The application of simulation depicts system performance with respect to change of variables. Chaar[7] reviewed different approaches and tools to developing manufacturing and real-time software. All these approaches stress the importance of formal models and specification languages in capturing functional and timing requirements of real-time embedded systems; and the role of prototyping in refining these requirements. However these approaches usually result in a static representation. Flexible process planning and scheduling methodologies are difficult to model using such formal approaches.

3. AN OBJECT-ORIENTED APPROACH TO FMC DEVELOPMENT

The FMC control software can be divided into three sets of major components according to its physical distribution and logical connectivity. These components include the cell supervisory unit, local control units, and database and information system. In particular, the local control units are modelled using a well defined object hierarchy (Fig. 1). The implementation of an intelligent shop-floor device (ISFD) is based on this hierarchy. These ISFD's are classified as processing (e.g. CNC machines, CMM, load/unload station); transferring (e.g. robot, pallet changer, tool changer); transporting (e.g. conveyer, AGV's) and storing. Using the concept of aggregation, any complex ISFD can be built upon the given basic devices.

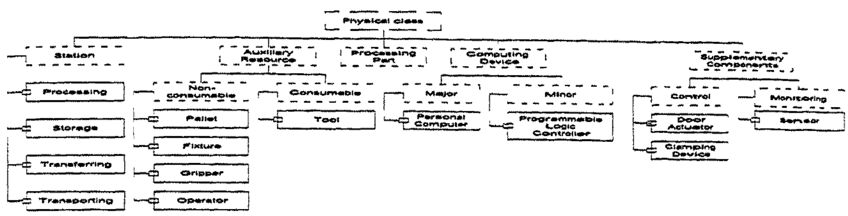


Fig. 1 Physical class hierarchy using object oriented modelling approach

The ISDF concept realises the goal of local intelligence so that control is distributed to the appropriate point of application. An ISDF is specialised in performing its own set of tasks autonomously. Its communication with the cell supervisor is confined to a small set of command and status reporting primitives. An ISDF also has the access right to necessary information from the database. It is worthmentioning that these ISDF's are generic software entities. They must be coupled with specific device drivers in order for complete installation and to exercise local control of shop-floor activities.

The object class consisting of such ISFD's allows the FMC control software to be developed without regard to the hardware details of the actual shop-floor devices. Such generic control software provides an effective solution to the software problem, especially when this concept is combined with the IDE design. Such an object-oriented FMC control software is highly reusable. However, the actual implementation of the software depends on the networking details and will be discussed in the following section.

4. A GENERIC FMC CONTROLLER ARCHITECTURE

4.1 Control hierarchy

In this discussion, a two-level cell-station control structure is adopted. A station here is defined as processing, storage, transferring, transporting or a combination of these capabilities. The structure employs a centralised control hierarchy where a cell supervisor is used to co-ordinate the tasks of all components in the cell. The major scheduling decisions concerning resource allocation, logistics and contingency planning are generated by the FMC supervisor. Stations in the cell, however, have a high degree of local intelligence especially in monitoring their own operations. In the proposed system, centralised control is preferred to decentralised control architecture. The structure has a number of advantages including the correspondence to corporate control hierarchy, simpler cell components co-ordination activities, lower communication requirement, easy adoption of different planning and scheduling methods, and more capable of generating optimal plans and schedules in real time.

4.2 FMC control software architecture

Essentially, the control software structure for FMC's should achieve hardware independence in order to improve its portability. Moreover, it is desired that such a structure will be highly modular to allow the software modules and various communication mechanisms for these modules to be developed in a separate manner. To achieve these goals, a three-layer modular architecture, composed of an operation layer, a networking layer and an operating system layer is designed (Fig. 2).

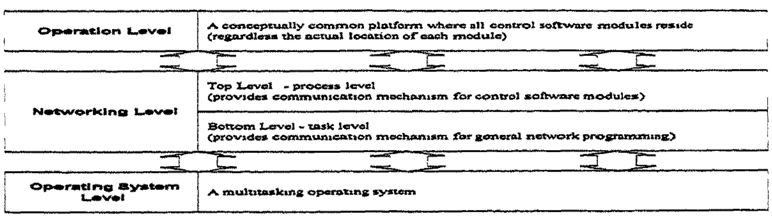


Fig.2 Three-level software decomposition hierarchy

The lowest level, the operating system level, consists of the drivers for various computing and communication hardware, operating systems and communication protocols. The operating system must be multi-tasking so as to enable the synchronisation of events as well as process co-ordination in real time. With respect to the operating system and communication protocols chosen, a fundamental set of I/O interfaces and communication functions are provided as a basis for the generation of control software.

The networking level acts as a linkage between the operation and operating system levels. It provides communication facilities for software entities at the highest level. The networking level is further decomposed into two sub-levels: the (FMC) process and task level. The task level is designed to interface to particular operating system and communication protocols chosen. It provides a comprehensive set of communication functions which are useful for general networking software development. By providing standard routines, communication to local or remote software modules can be treated in the same manner. Organising the networking level modules in this way achieves a clean logical structure of the software/hardware entities. Thus a conceptually homogeneous environment is provided for software development, where information concerning the location of each software module, communication mechanism used between different computing devices and for different types of information can be masked. On the other hand, the process level provides functionality which supports FMC applications. Dedicated functions are built up on functionality provided in the task level so that the advantages of having a homogeneous platform are still retained. These functions include equipment status and fault reporting, command and database related functions. Software reusability in FMC development is therefore encouraged. Moreover, meaningful and direct expression of functions allow easy modification.

The highest level, operation level, serves as a platform where all control software application modules reside. Modules embracing real-time and off-line processes, as well as cell and equipment control stay together. By functional decomposition, each module is embedded with distinct functional scope (Fig. 3).

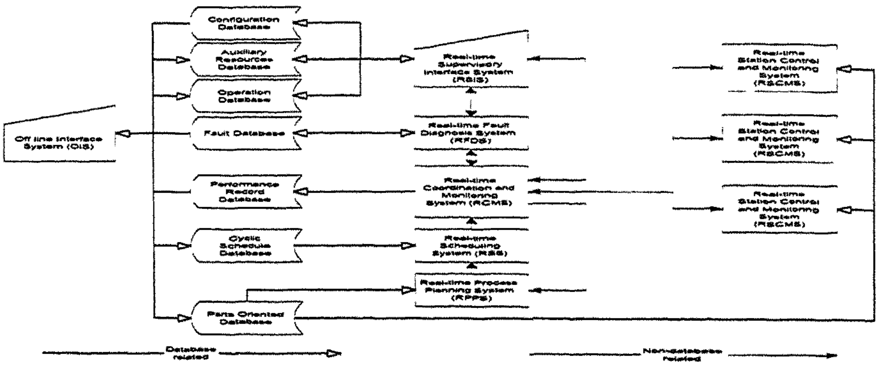


Fig.3 Functional block diagram in operation level

The three-layer hierarchy simplifies software development work by separating complicated networking and computing hardware issues from cell control functions. Through the use of device drivers to interface equipment, hardware independence is achieved. On the other hand, the establishment of networking level also improves software reusability and modifiability. Developers can adopt either the whole networking level functionality or one sub-level according to the system requirements. Such flexibility further encourages software reuse. Further, modularity and exact definition of interrelationships between software modules subsequently lead to high connectivity and enhance software re-use. Lastly, the separation of process planning, scheduling and control functions retains maximum flexibility in adapting different control strategies.

5. Integrated Development Environment

5.1 Typical system development process

The development of FMS's and/or FMC's consists of various stages including analysis, design, evaluation, implementation, deployment and evolution. In designing an FMS, both hardware and software issues should be considered concurrently in order to achieve a satisfactory solution. FMS developers need to account for the capabilities of individual hardware and software components. Also, the effects of combining those components must be considered. The design is therefore refined incrementally and finally a satisfactory solution will be obtained. Following the design stage, implementation including the preparation of system control software and installation of both software and hardware component can then be performed.

The provision of tools for the development of FMC's aims to improve the development process as well as to prepare for system evolution. Automating stages of development process is useful for improving development process.

Other tools like specification languages and formal models aim to avoid ambiguities in design. Evaluation and optimisation tools can also be extensively applied to identify inherent design faults, evaluate system performance and search for an optimum combination of software and hardware components.

5.2 *Integrated development environment*

An integrated development environment for FMC is a collection of development tools implemented to partially or fully automate FMC development process. The provision of development tools under the same environment has several advantages. Firstly, the situation of having a divergent set of databases can be avoided. Simple information management and maintenance can therefore be achieved. On the other hand, the integration of the development tools also represents compatibility of data format among these tools, where information generated by a tool in one stage can be used by another tool in the next stage directly or through simple automatic data conversion process, thus reducing duplicating effort and unnecessary errors.

The integrated development environment for FMC's (FMC-IDE) proposed here supports a majority of development stages from design, evaluation, implementation, testing to deployment. The FMC-IDE essentially consists of a system specification facility, source code generator and compiler, simulator and emulator (Fig. 4). The design of the FMC-IDE is based on the generic controller software architecture proposed. Default methods for process planning, scheduling and control are provided in the FMC-IDE. Software and hardware design details are entered into FMC-IDE through system specification facility. They can then be converted to program source code, and in turn to executable program for evaluation, testing and final installation purposes respectively.

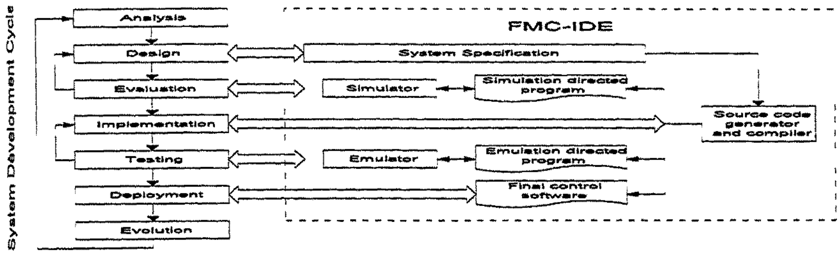


Fig. 4 Application of FMC-IDE among system development cycle

5.3 *Planning, scheduling and control strategies in FMC-IDE*

As mentioned in the above section, default process planning, scheduling and control strategies for control software developed are provided by the FMC-IDE. It is noted that developers can incorporate their own process planning, scheduling or control functions if they consider appropriate. However there is not any support for source code generation other than default those procedures.

Process Planning. There is not any default dynamic process planning functions provided. Under this arrangement, process planning work is carried out off-line. The procedure begins with transferring the process plans and machining files to the part oriented database (refer to Fig.2) from other sources (e.g. CAPP). System related information such as job assignment to station is then added to the plans through an off-line interface system. It is worth mentioning that the generic control software architecture still allows planning to be performed in real-time to enhance the integration of computer based process planning, and in turn CIM development.

Scheduling. The default scheduling method is a two-stage heuristics based improvement procedure. An initial schedule is generated first by applying heuristics using several criteria with a consideration of both system efficiency and priorities of individual stations and jobs. An improvement procedure using pairwise interchange is then carried out. As the solution space is enormous, the search procedure is performed according to the time limit and computing capability. It is also noted that the generic control software architecture provides capability to use cyclic schedule developed off-line. The advantages of cyclic schedule include: better schedule and less utilisation of computing resources in real time.

Control. The control strategy concerns mainly congestion problem in the cell and fault detection. Congestion is avoided by limiting the total number of parts in the system, and is relieved by using reserved buffer space. On the other hand, a fault is identified by comparing estimated processing times with real time performance. The type of fault and the related information on the station, auxiliary resources and parts are reported to the real-time diagnosis system for the investigation of the origin of the fault. At the same time, real-time process planning and other related system will be triggered to cope with necessary changes.

5.4 *Source code generator*

Source code generator translates details from specification facility to source code in C++ programming language. Three versions of source code can be generated for the purpose of system evaluation, control software testing and final control software generation. C++ is adopted for its popularity and object oriented nature. The generation of source code has the advantage of allowing further modification and insertion of alternative software modules to replace existing ones. To support the generation of source code, a library of data structures, classes and functions are prepared in advance. Object-oriented modelling (O-OM) approach is extensively applied to prepare the library of software components. Physical as well as information objects are modelled using O-OM to take advantages of modularity, flexibility, adaptability and disciplined methods. With the provision of software components, developers can alternatively generate their own program modules.

5.5 *Simulator and emulator*

The simulator is used as an evaluation tool in the FMC-IDE. It helps determine the correct configuration of hardware components; identify bottlenecks; estimate utilisation, work load balancing, work-in-progress level; and allow sensitivity analysis to be performed by repeating experiments with different parameters. The simulator acts as an experimental frame where combination of cell control software modules are assessed. It is configured according to hardware specification and simulation details such as fault probability, as well as warm up and simulation time. After thorough evaluation of design, control software can then be generated by source code generator and compiler. It is then tested using the emulator to guarantee normal operation. The emulators model the stations' behaviour, including fault and failure situations. All required emulators are installed locally and interfaced to corresponding local station control and monitoring systems respectively.

6. CONCLUSION

The paper describes a novel approach to the FMS software problem. Two new viewpoints are embodied in this development. The first is reviewing the role of the user in the FMS design-implementation-operation cycle while the second is the establishment of close linkages between these three phases in this cycle. The result of such ideas is highlighted by the implementation of an object-orientated FMC-IDE. Another design idea of separating the complicated networking and database related issues from control software design effectively speed up the software development process. Further, a modular architecture with separated planning, scheduling and control functional structure allows application of different strategies. It is found that the close linking of development tools naturally provides guidance to system development. Also, automatic generation of simulation model and control software speeds up the development process where effect is especially apparent in repetitive design-evaluation and testing-implementation cycles. Finally, the use of O-OM and provision of software libraries enhance software reusability.

REFERENCE

1 Wright P.K., "Principles of Open-Architecture Manufacturing", *Journal of Manufacturing Systems, Vol.14, No.3, 1995. pp.187-202.*

2 Levin L., Fielding E. & Ackhurst K. "Developing a Computer Control System for a Flexible Manufacturing Cell", *Int'l Journal of Flexible Manufacturing Systems, No.5, 1993, pp.143-159.*

3 Dolinska M. & Besant C.B., "Dynamic Control of Flexible Manufacturing Systems", *Intil Journal of Advanced Manufacturing Technology, No.10, 1995, pp.131-138.*

4 Nof S.Y., "Critiquing the Potential of Object Orientation in Manufacturing", *Intil Journal of Computer Integrated Manufacturing, Vol.7, No.7, 1994, pp.3-16*

5 Smith J.S., Joshi S.B., "Reusable Software Concepts Applied to the Development of FMS Control Software", *Intil Journal of Computer Integrated Manufacturing, Vol.5, No.3, 1992, pp.182-196.*

6 Talavage J.J. & Shodhan R ., "Automated Development of Design and Control Strategy for FMS", *Intil Journal of Computer Integrated Manufacturing, Vol.5, No.6, 1992, pp.335-348.*

7 Chaar J.K., Teichroew D. & Volz R.A., "Real-Time Software Methodologies: Are They Suitable for Developing Manufacturing Control Software?", *Intil Journal of Flexible Manufacturing Systems, No.5, 1993, pp.95-128.*

PROPOSITION OF AN APPROACH FOR A PHYSIO-ECONOMIC PRODUCT EVALUATION, STARTING AT THE DESIGN PHASE IN A CONCURRENT ENGINEERING CONTEXT

D.Raviart, O.Sénéchal, C.Tahon
Université de valenciennes et du Hainaut Cambrésis
Laboratoires d'Automatique et de Mécanique Industrielles et Humaines (France)
(LAMIH - URA CNRS n°1775)
Tel: 33 27 14 13 54 E-mail: draviart@univ-valenciennes.fr

ABSTRACT:

To implement concurrent engineering implies an important evolution in the decisions that each participant must make for the realisation of one product. One some desired objectives is to evaluate the realisation of a product, beginning in the design phase, in order to choose the most efficient solution in economic, temporal, and qualitative terms, without loosing sight of the internal environment and the internal and external problems of the firm.

Our approach consists of integrating the evaluation process in a concurrent design cycle. This evaluation begins with an estimation of different solutions for the realisation of a product in economic, temporal and qualitative terms. In the second phase, the various solutions are compared in terms of diverse criteria (money, time, quality).

KEYWORDS:

Concurrent engineering, manufacturing processes systems, decision support systems, multicriteria approach, activity.

1. INTRODUCTION:

The fierce competition resulting principally from the commercial accompanied exchanges is today accompanied by consumer demands in terms of keep same order as above price, delays and quality. In less than twenty years, the world has gone from a production economy where the demand superior supply, to a market economy what there is a differentiation between the firms "par excellence" (6). In this context, staying competitive has become a multicriteria problem. Controlling cost, quality and delays is essential in order to maintain this competitiveness.

In the first part of the paper, we present the objectives of the evaluation and the place of the evaluation in a conception concurrent cycle. In second part, we propose a evaluation methodology for the design phase. Finally, we using an example, we show the evaluation could be implemented in the firm.

2. THE OBJECTIVES OF THE EVALUATION

A firm must stay competitive in order to reach its social objectives, the contribution at the good well-being; financial objectives, the creation of profits; and economic objectives, the distribution of wealth. This competitiveness is directly dependent on the company's control of internal production factors (2). Indeed, a competitive firm must design a product which satisfies the economic and function satisfactory needs of the users while using the available firm resources in an optimal way. To reach these objectives, the firm unavoidably passes through a phase of analysis where the company's requirements in terms of production, human, materiel capacities, task complexity as well as price and time considerations are examined. In these conditions, the evaluation must constitute a means of identifying the best production procedure within the framework of decisions which are as much political as strategic and tactical.

3. EVALUATION IN A CONCURRENT DESIGN CYCLE

In a process of evaluation destined to limit the inherent risks of beginning to manufacture a new product, some conditions must be respected:

- the evaluation must take place in phase the design since 80% of production costs are define at this stage (3). The limitation of the risks incurred during of the design stage can not be obtained without integrating manufacture constraints as much as possible when making design decisions and estimating the production costs, and the delays as early as possible and in a reliable manner,
- the capacities , financial material and human must be well known,
- the functional features the product must be clearly defined,
- experts in all the area touched by the life cycle of product must participate in the evaluation,
- modelling and simulation tools, enabling the organisation of these competences and the observation of the effects of the decisions made, must be available.

The above conditions require that the evaluation, be integrated in a concurrent design cycle. It is for this reason that the technical choices relating to the product and at the process will be made conjointly so that the economic, physical and functional consequences can be established. In the problem which interests us, the concurrent conception is at the same time a necessary condition and an objective for disposing of technical data regarding the process and the product, and is indispensable in order to achieve a valid evaluation. But the result of this evaluation, and thus the functional, organisational and social data allow the reactivate on of the design cycle of the product, the process, or both, or its continued development up to in the start of production (cf.figure1).

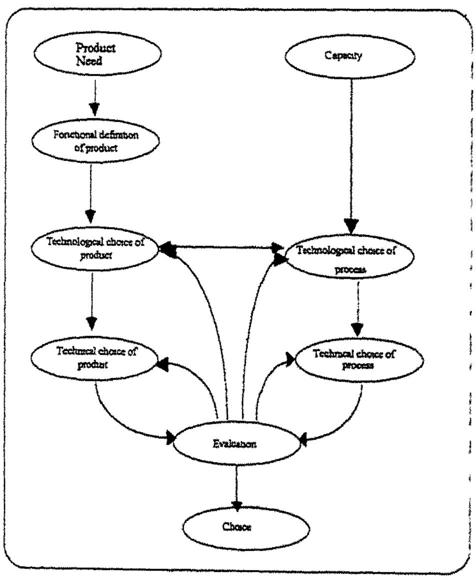


Figure 1: The evaluation in the concurrent cycle of conception (10)

In the following section, we propose an evaluation methodology for the design phase.

4. PROPOSITION ONE METHODOLOGY OF EVALUATION OF THE CONCEPTION PHASE

Evaluation means assigning a value good or bad, better or worse, at an entity or an occurrence (2). We use this definition to situate our research in the problematic of evaluation in the design phase. We consider that evaluation consists, in the first phase, of estimating the physical and economical performances of one suggested solution, and in the second phase, of comparing these performances in order to make a choice from among all the suggested solutions (cf.figure2).

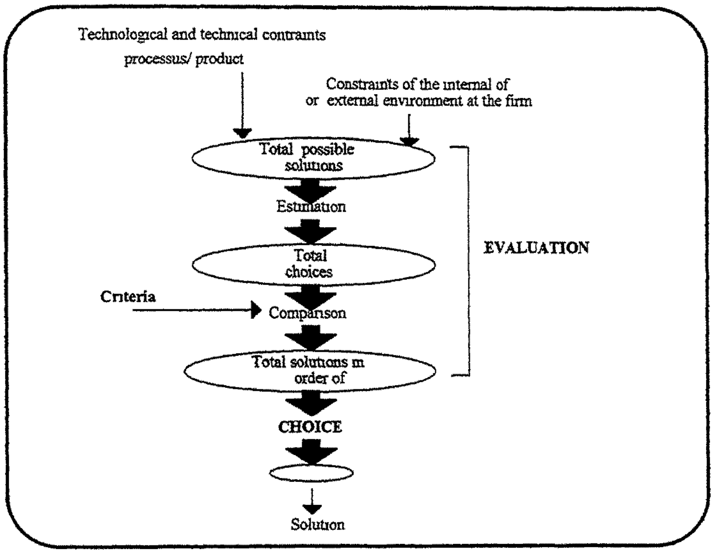


Figure 2: Presentation of the evaluation and the choice

We present now the evaluation function composed of the estimation function and comparison function.

4.1 L'estimation

The methods of cost estimation during design (6) used today don't permit a correct decision concern the realisation of a product. To made, because these methods don't allow the decision maker to integrate the risks that the firm, or the indirect costs that the firm generates for example. Finally, the methods of cost evaluation don't contribute to a realistic evaluation of the various scenarios for the realisation of a product.

We approach at present the base model : the activity that we used in the function of estimation.

The base model: The activity

To implement the estimation function, it is necessary to dispose of models which allows the integration of the firms characteristics, for example its know-how or its physical systems. Our contribution consists of an approach of support modelling and simulation for estimations based on the activity concept.

The qualities of the concept of activity in the context of evaluation are follows (5) (1):

- the activity is coherent, with a vision in term of firm know-how and of competence,

- it is relatively robust with regard to the frequent changes die to the instability of the environment,

- it offers a common base for the measurement via the physico-operational indicators performance.

This article don't give detailed description of the estimation function implemented to the determine the features of physical and economic performances in the paper. Many papers describe the principles of the estimation function (11) (12) that we introduced briefly. We approach at present the comparison function.

4.2 The comparison

The comparison function allows us to evaluate the different configurations with regard to the physical and economical performances of the configurations estimated for the realisation of a product.

In order to provide a support for decision-making in the context of concurrent engineering, we suggest the use of the multicriteria decision methods (9) (13) (7) which resolve at the comparison problematic of comparison.

These decision making methods enable:

- the choice one configuration of realisation one product in function of various criteria,
- the integration of the human operator,
- the qualitative parameters (For example: product quality,......).

We defined 4 four families of criteria used by the comparison function who are the criteria:

- criteria of delays,
- criteria of validity one configuration,
- criteria of quality,
- economical criteria.

These four big one criteria families permit at the decision maker of finding again its own laws of decision.

We approach to present some observations resultant to implement of the first stages of the process on an industrial site.

5. IMPLEMENTATION ON AN INDUSTRIAL SITE

The site selected in order to validate our approach is a manufacturing firm working for the aeronautics. The production is there very little standardised. It employs about 350 persons and works principally at the orders.

5.1 Analysis an research

The first procedure in the firm consisted in to identify the activities and to define the physical and economical associated features. Most of the information are collected by utilisation the tools used in the methods of analyses by the activities, that is:

- the questionnaires and interviews

- the measures in the workshops and analyses of historic.

The first remark that will make us on the modelling in activities is that the majority of the information regarding the activities of manufacture, control, and assembly exists in the methods office of the firm. It brought us to analyse operation list for the determination of the different operative activities .

To implement our approach required the motivation of personal of the firm, persons responsible or "operational." It is passed by a first phase of presentation of the objectives and of motivation of personal and it was introduced at the persons like a project of firm which the success depended of their implication.

5.2 To implement of evaluation

The elements of this to implement are the following:

- the modelling in process implicating the persons responsible and the "operational" of the firm,
- the research of information at the level of each activity (time of cycle, number of product, cost amortisation machine,...). These different information are descended of information system or by the study of each work station of the firm,
- the calculation of cost and the determination of the activities to excessive cost,
- the proposition new operative scenarios in order to achieve an reliability evaluation in function of various criteria. It is achieved with the "operational" of sector of production.

6. CONCLUSION

We introduced the objectives of the evaluation and the evaluation in a design concurrent cycle since the design phase of product. We suggested a methodology of evaluation composed of estimation function and comparison function and an example of to implement.

The first conclusions that allows us this to implement carries on the efficiency of the method in term of integration of the competence and of personal motivation in the involvement at the evaluation.

7. REFERENCES

1. BRINSON J.: Activity analysis worshop, conference CAM-I, Munich, 11 décembre 1990.

2. JACOT J.H. : A propos de l'évaluation économique des systèmes intégrés de production. ECOSIP, Economica 1991.

3. JAGOU J. : Concurrent Engineering, la maîtrise des coûts, des délais et de la qualité. HERMES 1993.

4. LEVINE P., POMEROL J.C : Système interactif d'aide à la décision et systèmes expert, HERMES, 1989.

5. LORINO P. : Le contrôle de gestion stratégique. La gestion par les activités. Dunod 1991.

6. MARTY C. : Le Mieux Produire. Approche productique du Management, Collection Lavoisier TECH & DOC , 1991.

7. MICHALES J.V., WOOD W.: Design to cost, Interscience Pubication, Edition Rodney D. Stewart, 1989.

8. POMEROL J.C : Choix multicritère dans l'entreprise, HERMES 1993.

9. ROY B., BOUYSSOU D. : Aide multicritère à la décision: méthodes et cas, Economica Gestion Paris, 1993.

10. SENECHAL O., LENCLUD T., TAHON C. : Production costs identification using value engineering and simulation in a concurrent engineering process. IFIP international conference "Feature modelling and recognition in advanced CAD/CAM systems" , May 1994.

11. SENECHAL O : Proposition d'une méthodologie pour l'aide à l'estimation des performances physico-économiques des systèmes de production dans une approche concourante. Thèse de l'Université de Valenciennes, 1996.

12 SENECHAL O, TAHON C.: A methodology for integrating economic criteria in design and production decisions. Ninth International working Senimar on Production Economics, 19-23 February, 1196.

13. TABUÇAMON T.: Multiple criteria decision making in industry, Elsevier, New York, 1988.

SYSTEMATICAL APPROACH FOR THE PRODUCT DESIGN

Jean TRIOULEYRE
Department of Engineering Production
Ecole Nationale Superieure d'Arts et Metiers
PARIS (France)

ABSTRACT

The global system of the product design includes three main components : product, interventions and environment. In this paper we examine more particularly the sub-system « product ».

Product is founded on three sub-systems : organization, valuation and typology. Organization concerns architecture and properties (material, form, dimensions and surface). Typology is characterized by many attributes. Valuation is representative of compatibility of the product considering feasibility, functionality and « marketability ».

The interactions of organization, typology and valuation give the basic rules concerning the selection of technological possibilities for the design product.

This presentation has a generic character.

KEYWORDS

Architecture. Basic technical properties. Typology. Valuation. Intertractions.

1. INTRODUCTION

The product design is a system in which we find three conponents which are : product, interventions and environment. For effect the analysis of the product design it is necessary to consider more particularly the product sub-system which is compounded of three components : organization, typology and valuation.

We propose to clarify interactions between these components for give an exhaustive list of possibilities in the demarch concerning the building of the product organization

2. THE PRODUCT ORGANIZATION

Considering the operative part of the product, we analyze the architecture and the basic technical properties.

2.1 The architecture

Architecture is founded of elements more particularly characterized by :
- the number,
- the nature of connections.

Architecture presents two limit cases :
- Monolithic architecture (one element). It is the maximum degree of the function integration.
- Compounded architecture (n elements). We obtain the modularity taking into account the auxiliary functions.

2.2 The basic technical properties

Each element is completely described by four basic technical properties : material (M), form (F), dimension (D), surface (S).

The structural analysis of components inserted in the organization system shows possibilities of the combinatory on which it is possible to simulate the constraints and the opportunities generated by them.

	ARCHITECTURE	M	F	D	S
ARCHITECTURE	X	X	X	X	X
M		X	X	X	X
F			X	X	X
D				X	X
S					X

Case 1 The first problem is to define separately the characteristics of possible materials, shapes, dimensions and surface, taking into account the functional specifications.

Case 2 If monolithic architecture, the first problem is the duality between basic technical properties.

Case 3 If compounded artchitecture, one problem is the compatibility between each element and their basic technical properties in the connection zone.

3. THE PRODUCT TYPOLOGY

The product typology concerns all attributes which have a significant implication considering decisions in relation to the manufacturing, the marketing and the utilization of the product.

3.1 The attributes

The following aspects can be considered.

General aspects

- Quantity
- Dimensions
- Duration
- Accuracy

Utilization

- Destination (automotive,aeronautic, machine-tool)
- Particular conditions of use (hygien, security...)
- Individual or collective use.

Manufacturing

- Conventional or non conventional processes
- Production rate

Marketing
- Mode of delivery
- After sale service

The structural analysis of attributes shows possibilities of the combinatory.

	Quantity	Dimensions	Accuracy	Production rate
Quantity				
Dimensions				
Accuracy		X		
Production rate	X			

<u>Case 1</u> Combination between the quantity and the production rate is a criterion for the selection of equipments.

<u>Case 2</u> The conjunction of big dimensions and high level of accuracy can introduce difficulties of manufacturing.

3.2 Interactions between typology and organization

The structural analysis of attributes inserted in the product organization can show possib ilites of the combinatory on which it is possible to simulate the apportunities and the constraints generated by them

ATTRIBUTES / ORGANIZATION	QUANTITY	DIMENSIONS	ACCURACY	PRODUCTION RATE
ARCHITECTURE		X		
MATERIAL			X	
FORM			X	
DIMENSIONS			X	
SURFACE			X	

<u>Case 1</u> Generally, big dimensions are associated to compound architecture, excepted when manufacturing process allows to obtain monolithic product (wire extrusion).

Case 2 Generally, high production rate is incompatible with big dimensions because specific problems in relation to transformation, and logistic processes.

Case 3 Accuracy combinated with basic technical properties is a determinist factor to increase quality but also constraints in manufacturing.

4. THE PRODUCT VALUATION

The valuation criteria allows to appreciate the appropriate characteristics of the product in relation to manufacturing, marketing and utilization.

4.1 The valuation criteria

The following aspects can be considered :

Manufacturing
- Formability
- Weldability
- Manufacturing cost

Marketing
- Price

Utilization
- Reliability
- Maintainability

4.2 Interactions between valuation and organization

The strutural analysis of criteria in the product organization can show possibilities of the combinatory on which it is possible to simulate the opportunities and the constraints generated by them.

CRITERIA / ORGANIZATION	FORMABILITY	CONNECTABILITY	MANUFACTUGING COST	RELIABITLITY	MAINTAINABILITY
ARCHITECTURE		x	x		
MATERIAL	x		x		
FORM	x		x		
DIMENSIONS	x		x		
SURFACE	x		x		

Case 1 Compounded architecture is directly associated with « connectability ».

Case 2 Material is directly influenced by feasibility (formability, for exemple).

Case 3 More generally, feasibility is simultaneously influenced by material, form, dimension and surface.

Case 5 We constat that the manufactring cost is influenced by several factors not only represented by the product organization. In this case, a largest approach is necessary.

This lat remark shows the necessity to realize a structural analysis taking into account others aspects in relation with technical, economical, organizational and human factors.

5. CONCLUSION

This schematic presentation of the systematic inventory of possibilities for elaborate the product design shows difficulty to regroup the significant factors because diversity of them.

However, many data bases are elaborate and particularly concerning manufacturing processes.

6. BIBLIOGRAPHY

J. TRIOULEYRE - Forming processes. Delagrave Editors. Paris 1984
J. TRIOULEYRE - Valuation criteria and design produt. 4th Congress Marseille 1993
J. TRIOULEYRE - Organization of knowledge and design product. CAD GRAPHIC'S WUHAN 1995.

FRAMEWORK TECHNOLOGY FOR TOOL INTEGRATION IN INTEGRATED PRODUCT DEVELOPMENT

K. Bender, K. Bindbeutel, M. Glander, A. Karcher
Institute for Information Technology
Technical University of Munich
Munich, Germany

ABSTRACT

The process of developing complex products within distributed teams of engineers using different methods and engineering tools on heterogeneous platforms is getting more and more complex and hard to manage. The research done within SFB 336 at the Technical University of Munich concentrates on the integration of product design processes and assembly planning processes.

The Framework technology has been selected as the platform for the integration concept to support Concurrent Simultaneous Engineering (CSE) in the product development process. Frameworks allow the defining of activities, tools, and datatypes to support task oriented use of development tools, thus reducing the administrative overhead typically involved with document oriented systems.

KEYWORDS

DATA INTEGRATION, TASK INTEGRATION, TOOL INTEGRATION, PROCESS INTEGRATION, FRAMEWORK, DESIGN MANAGEMENT, FLEXIBLE ENGINEERING ENVIRONMENT

1 INTRODUCTION

The Special Research Programme 336 (SFB - Sonderforschungsbereich) is a consolidation of six departments at the Technical University of Munich, which has been sponsored by the DFG (German Science Foundation) since 1989. This research area involves developing methods for computer-supported tools, as well as the development and operation of these tools. The goal is to support and improve the entire production process from the construction and assembly planning to the realization of a flexible assembly cell.

The authors of this contribution are in the process of organizing a new sub-project within this SFB. This project investigates the applicability of frameworks in conjunction with the integrated product development. The present status of integration has been reached using a central data base. The current research is focused on an integrated development environment, which flexibly allocates tool functionality to users. This development environment has to check for data consistency including version and variant management. It should also monitor the processes with regard to consistency and check if the data is up-to-date.

This contribution is organized as follows: first a detailed goal description of the SFB and an analysis of the specific characteristics and requirements of the integrated product design and assembly planning processes, followed by a short description of common framework systems initially applied in other areas (e.g. software engineering, electronic CAD), a more exact description of the framework technology, and finally the concept of an architecture combining the process oriented with the conventional data oriented approach.

2 CE-RELATED CHARACTERISTICS OF INTEGRATED PRODUCT DEVELOPMENT

Fundamental processes that designers and assembly planners use in product development can be divided into the phases *problem definition, modeling, design* and *realization* [2]. The most important outcome of the problem definition is a list of requirements, which specifies the desires and demands on the product to be developed. In the second phase, a function structure is established based on the requirements. The overall function of the product is broken up into smaller sub-functions, which are mapped to elementary functions, e.g. *direct*, *save*, *separate*, and *change*. Physical solution principles are then sought after for the elementary functions and combined in various ways to form model variants. During the design phase, the different variants can be compared with each other and the best one is chosen for further development.

The assembly planning process can be divided into the phases *generation of the operational plan, definition of the joining processes, resource selection and/or design* and *layout of the assembly cell.* A sequence of required assembly steps is defined in the operational plan according to the product structure, which is defined by the designer. First, the required joining processes are tested for feasibility (by means of simulation). Then, the various steps are assigned resources which are parameterized, programmed or newly designed. This finally results in the first layout of the assembly cell, which will be fine tuned in a subsequent optimization cycle.

The assembly planning process runs parallel to the design process (see Figure 1) [1,9]. Properties relevant to assembly are defined according to the degree of realization of the product design. By structuring the design into functional groups, initial cost and resource estimates can be made based on experience. As soon as the geometric information is available for components and assemblies, the required joining processes can be planned and initial design decisions concerning the necessary resources can be made.

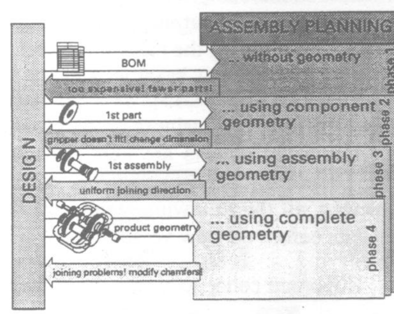


Figure 1: Integrating design and planning

To shorten the total development time, the specific tasks of engineering and assembly planning have to be done in parallel, working with a common data model. The consistency of this common data model has to be supervised by a process oriented component, organizing the interdependent sub-tasks. This approach is still prevented by the predominant *engineering islands*, meaning that a few engineering tools are grouped together allowing data exchange to some extent but offer a very limited capability of data exchange with other engineering islands. The solution to overcome these isolated islands has to focus the overall process of product design, since joining these islands using some appropriate data interchange format would simply multiply the various versions of each document.

3 THE FRAMEWORK ARCHITECTURE

One approach for solving problems involving complex processing steps, which are distributed in space and time among many workers, are *frameworks*. Initially they were established to coordinate the development processes found in electronic CAD. Frameworks help in precisely coordinating the processing steps, which are becoming increasingly interwoven. To be able to guarantee consistent data, “reservation mechanisms” are necessary. These guarantee a serialization of the process steps by reserving the data entities with different granularities. When many different tools work together, there is the additional problem of dealing with data specific to a version or variant of a tool. The data must be kept consistent for each tool, to allow processing, and must then be converted into the tool-specific format.

One important aspect to consider for different tools working together is the support of process or development logic. This means that certain development steps can only commence after other steps are complete, since the necessary data is not available yet or has not been up-dated. How strong this data relies on this information depends on the development process. The different workers must also be coordinated.

To take over these development areas frameworks incorporate a *data management* component, which either is based on an internal or external data base, and a *design management* component, which makes it possible to model development processes schematically in the framework. The third basic component includes mechanisms to perform flexible *tool integration*, meaning the assigning of commercially available tools to specific tasks.

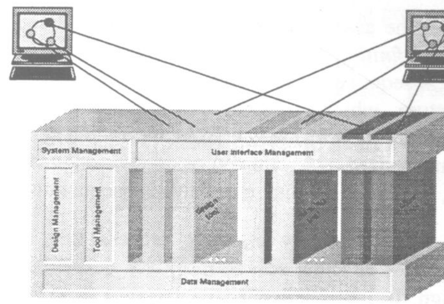


Figure 2: Modular structure of frameworks

Different frameworks have been developed in several research projects concerning Tool Integration in SW-Engineering and Integrating processes and tools in Electronic Design Automation ([3, 5]). The commercially available framework SIFRAME[1]. — as an output of these research projects — is building the basis for the research efforts described in this paper.

The core of SIFRAME is a data base, which stores the design objects to be processed, as well as representations of business processes to be handled. *Data management* makes sure that the data of the manipulated objects are kept consistent. One of the main features of a framework is *design management*, which makes sure that valid input data is readily available for a tool invocation. If data is changed, all the following results are marked as invalid and the process restarts with the changed input data before the new version of the sub-project is completed.

Important concepts which are to be realized by the *design management* of SIFRAME are:

- *access control* to objects with users and their roles
- *version management*: creation and storage of different design object versions. New versions can be derived from existing ones or can be newly generated.
- *flow management*: monitoring the execution sequence of activities. The validity of input data is checked before the tool is invoked.

[1] SIFRAME is the product name of the framework that originates from the JESSI Common Framework [9]. This framework has originally been further developed by Siemens-Nixdorf Information Systems and is now put on the market by Siframe Software Technologies GmbH, Munich, Germany.

- *design decomposition*: project structure; Division of a project into tasks. Parent-child relationships (hierarchy and network). Such a relationship implies that a task needs the results of one or more sub-tasks.
- *object status*: design objects have an up-to-date process status which controls which activities may be executed by which users. A publish mechanism releases processed objects. These objects can then be used by other tasks and people.
- *variant management*: the execution of an activity generates a result which is usually overwritten after each further invocation. To prevent that, variants of design objects are produced. The variant management generates and handles the variants. A variant is an instantiation of a flow, therefore containing representations of activities, including their results. A user can generate and try out as many variants as he wishes, but must decide which variant to publish as the single valid version.

Data management and *design management* provide an assistance in enabling project work to be done in parallel by different people, which is known as Concurrent Simultaneous Engineering (CSE).

4 CONCEPT FOR FRAMEWORK-BASED INTEGRATION

The transformation of the concept of design management, supported by frameworks, to engineering processes can be described with the term *task-integration*. The framework serves as a domain-neutral platform that integrates the different tools, provides the data needed by the tools and permits the definition and coordination of the domain-specific processes that follow the interlocked structure described in chapter 2.

The framework thus provides a *process-oriented* view. Processes consist of separate tasks, linked into process-chains. This means that the task to be worked on has already been defined, and the framework provides the necessary data and software-tools, whilst checking for inconsistencies in the process-chain: one cannot work on a task unless the former tasks have been completed.

Figure 3 illustrates at the left side an example of the basic concept of design management: *project „Change"* as a collection of *tasks*, each of which incorporates a specific flow representing the engineering-specific chain of *activities*. An *activity* represents the indivisible usage of a tool using and creating specific documents in the context of that task. The cooperation of these framework functionalities allows data to be kept consistent for all tools and monitors the elementary process logic. The right side of Figure 3 shows a corresponding screenshot of SIFRAME™ with a concrete process model at the task level.

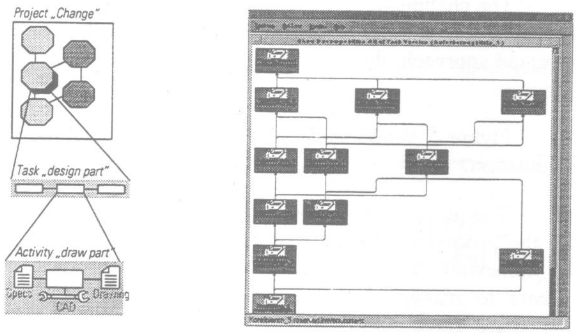


Figure 3: Design management

This approach is suitable when the natural way to do things is task-oriented and can be described in a top down decomposition starting from the highest level of the complete project and then breaking it down into tasks, sub-tasks and activities. The mapping of this basic concept to the iterative structure of the interrelated and interlocked tasks and sub-tasks of product design and assembly planning (see chapter 2) however leads into a specific problem domain:

To be able to utilize the principals of the design management with a deep semantic relationship to the application it is necessary to obtain a detailed objective of the exact requirements of the design and assembly planning process. This leads to a high granularity in the process description and to an exact definition of the product structures as well as the dependence on assembly specific information in the process model. At a very early stage the hierarchic structure of design management necessitates a detailed description of the sub-processes and activities in the integrated product development cycles. The engineering process however does have a slight reciprocating character with respect to the product structures. This leads to a contrast between the process-oriented view of frameworks and the structure-oriented view of data and information arising from the product development cycle.

On the one hand there is a detailed decomposition of the engineering processes towards a fine granular „project-task-activity-tree“ which makes a maximum semantic support by the framework design management functionality possible. On the other hand however this approach leads to an implicit mapping of the very complex product structures and requires, that all the different process relationships between the product parts, assemblies and subassemblies can be described and modeled at a rather early development stage. This however is a contradiction to the flexible, iterative and somehow intuitive way engineers want to design there products. One cannot focus on the engineering process and disregard the product model data, since there are tasks which can be performed best browsing the product model data and deciding *ad hoc* which pieces of data to change and how - an approach that opposes every process oriented method.

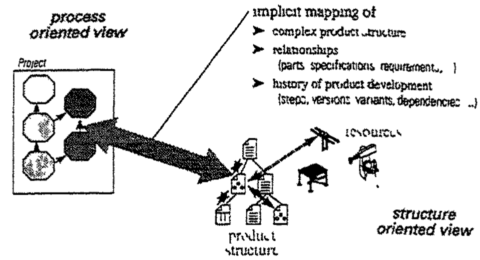


Figure 4: Process vs. Structure

This structure oriented view to this kind of engineering processes is typically addressed by EDM/PDM systems ([6]). They visualize the generalized product-model structure allowing the engineers to navigate within a *tree* or rather a *net* of all related product-defining items like assemblies, parts, variants and all the related documents.

The challenge thus is to combine these two requirements: process oriented approach, where the user activates a specific task that implicitly identifies the necessary data as opposed to the data oriented approach, that lacks any supervising structures to ensure data consistency in the engineering process.

Figure 5 shows the 2-layer concept of combining process orientation and product structure orientation:

The upper level, addressed by the process modeling paradigm of framework design management, allows to manage and coordinate the more fixed and standardized engineering workflow pattern within a team of engineers and a landscape of different tools all integrated to one homogenous engineering platform. Starting at a specific level of granularity in the product structure there is the lower part of the workspace- mapped to

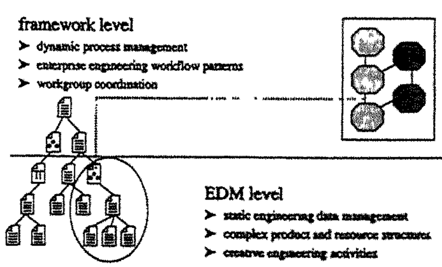


Figure 5: Two-layer concept

the functionality of a standard EDM-System - that is still „under construction“ and therefore subject to frequent changes and iterations. The responsible engineer (or group of engineers) are therefore free and also responsible to reorganize this part of the product-defining data outside the close controlling of the framework design management functionality.

It still has to be investigated, what in the field of integrated product design and assembly planning is the right level for this separating line between the two layers. This may also depend upon company-specific standards and decisions. Our goal is to develop a flexible concept that allows users themselves to define the needed grade of fixed and therefore controllable processes and more intuitive engineering tasks respectively.

5 SUMMARY

The integration of design and assembly planning in mechanical engineering can be seen as a typical example of integrating two interdependent engineering processes. Two fundamental approaches are: data oriented on one hand versus process oriented on the other. It could be shown that neither of these approaches for itself is the ultimate solution, since the way to accomplish a specific task depends on the nature of that task. Sometimes, it is more appropriate to identify the data needed and — data dependent — use the tools associated with that task, while some other tasks have to be defined in the context of preceding and succeeding tasks to ensure data consistency.

The approach described here addresses this by combining two technologies, each specialized for one of those aspects. PDM systems are close to being the de-facto standard for modern data management solutions, offering a data oriented approach. Every user has the ability to browse data structures and relations to identify the data he needs. The process oriented approach on the other hand is the central feature of framework-technology. Both systems have to be integrated by defining an „invisible“ horizontal line, separating the fields of influence to reach a higher level of integration.

6 BIBLIOGRAPHY

[1] *N.N.:* Sonderforschungsbereich 336, Kolloquiumsband vom 28.04.95, Technical University of Munich, 1995

[2] *Ehrlenspiel, K.:* Integrierte Produktentwicklung; Hanser Verlag München Wien, 1995

[3] *N.N.*: Standard ECMA-149: Portable Common Tool Environment (PCTE) Abstract Specification, ECMA 1993

[4] *Oliver, Huw:* Adding Control Integration to PCTE, Hewlett Packard 1991

[5] *N.N.:* Standards for Electronic Design Automation - Computing Environment Services; CAD Framework Initiative, Austin (USA), 1992

[6] EDM-REPORT Nr. 1 1995 - Sonderheft des CAD-CAM-Reports, Dressler-Verlag 1995

[7] *N.N.:* SIFRAME V3.0 System Overview Edition September 1994, Siemens-Nixdorf Informationssysteme AG, 1994

[8] *Feldmann, C.; Cuiper, R.*: Simultaneous Engineering in der Montageplanung. In: Reinhart, G.; Milberg, J.: Planung von Montageanlagen; Herbert Utz Verlag Wissenschaft München 1995, S. 1-18. (iwb-Seminarberichte (1995) 13)

[9] *Liebisch, D. C.; Jain, A.*: JESSI COMMON FRAMEWORK Design Management – The Means to Configuration and Execution of the Design Process. IEEE (1992) 8, 552 - 557.

INTEGRATION OF DESIGN SOFTWARE FOR CONCURRENT ENGINEERING

Ke-Zhang Chen
Department of Mechanical Engineering
The University of Hong Kong
Hong Kong

ABSTRACT

Concurrent Engineering can help manufacturing enterprises to achieve shorter time to market, reduced development costs, and high quality products. In order to realize the concurrent engineering, it is necessary to integrate the people with different disciplines, design methods, design tools, design data, and so on properly. This paper discusses the integration of software of existing design methods for concurrent engineering by using Axiomatic Design. The results show that a very complicated software system for concurrent engineering becomes very simple and has n modules corresponding to n design methods and one main module which contains all the junctional properties at each level.

KEYWORDS

Concurrent Engineering, Design Software, Axiomatic Design, Design Theory and Methodology

1. INTRODUCTION

For the world-wide competition, more and more manufacturing enterprises have adopted the concept of concurrent engineering as a strategy for achieving shorter time to market, reduced development costs, and high quality products. Concurrent Engineering requires the effective integration of the people with different disciplines, design methods, design tools, design data, and so on. It means that the organizational structures must be changed and computers must be used with the integrated software system for concurrent engineering since the database for it is too large for human designers to handle efficiently. In the previous research[1], it was concluded that Axiomatic Design[2] is only one method which illustrates clearly the design principle, design process and design method, and is an effective design framework. Therefore, Axiomatic Design should be used to guide the effective integration. This paper discusses the integration of software of existing design methods for concurrent engineering by using Axiomatic Design.

2. AXIOMATIC DESIGN

According to the theory of Axiomatic Design[2], design involves the continuous processing of information between and within four distinct domains: consumer domain, functional domain, physical domain and process domain. The needs of the customer are established in the consumer domain and then are formalized in the functional domain as a set of functional requirements (FRs) that govern the solution process, where each FR is independent of other FRs by definition. The solution alternatives are created by mapping the FRs in the functional domain (What we want to achieve) to a set of design parameters(DPs) in an adjacent physical domain (How we achieve them) which are then mapped into the process domain in terms of the process variables (PVs). The output of each domain evolves from the abstract concepts to the detailed information in a top-down hierarchical manner. The hierarchical decomposition in one domain cannot be performed independent of the evolving hierarchies in the other domains. The decomposition thus follows zig-zag mapping between adjacent domains. The number of plausible solutions for any given set of requirements depends on the imagination and experience of the designer. Then, the design axioms are used to determine the acceptable designs. The first axiom(Independence Axiom) states that a good design should maintain the independence of functional requirements. The second axiom(Information Axiom) establishes information content as a relative measure for evaluating and comparing alternative solutions that satisfy the Independence Axiom, and states that a best design should have the minimized information content.

The mapping between two domains, e.g. the functional domain and physical domain, can be expressed mathematically as a design equation:

$$\{FR\} = [A]\ \{DP\} \qquad (1)$$

where {FR} is a vector of functional requirements, {DP} is a corresponding vector of design parameters, and [A] is the "design matrix" that defines the nature of the relationship between each functional requirement FR_i and each design parameter DPj. The elements of [A] are given by:

$$A_{ij} = \frac{\delta FR_i}{\delta DP_j} \quad (2)$$

If [A] is a diagonal matrix, i.e. a square matrix whose non-zero elements only occur along the main diagonal, the design satisfies the Independence Axiom and is called an uncoupled design. If [A] is a triangular matrix, i e. a square matrix whose non-zero elements occur in a triangular pattern either above or below the main diagonal, the design is a decoupled design and also satisfies the Independence Axiom, provided that the DPs are changed in a specific sequence such that each FR_i is ultimately controlled by a unique DP_j. If [A] is any other type of square matrix, the design is called a coupled design and does not satisfy the Independence Axiom so that it is not acceptable.

A design's information content is calculated according to the following logarithmic expression:

$$I = \log_2 (1/P) \quad (3)$$

where p is the probability of successfully satisfying the functional requirements. The probability of success is a function of both the design range and the capability of the proposed solution.

3. SOFTWARE DESIGN BASED ON AXIOMATIC DESIGN

According to Axiomatic Design, software design, like other product and process designs, is a successive interplay between domains. When the decomposition process propagates down to the lower levels, a designer can reach the level where one FR can be fully satisfied or controlled by the selected set of DPs and does not need to be decomposed any further. It forms a terminal node of the hierarchical tree. The decomposition process terminates when all the branches of the FR tree form the terminal nodes. Thus, each terminal node has one module of software which can be readily coded into a program and transforms a set of inputs to the output. Since the modules are generated to make the associated FRs uncoupled or decoupled, the software system designed in this way has good modularity. Therefore, software design based on Axiom Design can facilitate system integration and immediate access to current users and existing software. The hierarchical tree structures of FRs and DPs and the analysis results in the form of a design matrix can be represented in a single diagram, as shown in Fig.1, which was developed and defined as the "Module-junction structure diagram" by Sun-Jae Kim et al[3]. It is composed of modules and junctions which represent FR terminal nodes and their vertical integration of child modules into a parent module respectively. Thus, a software system has n modules corresponding to n FR terminal nodes and one main module which contains all the junctional properties at each level, as shown in Fig.2[3].

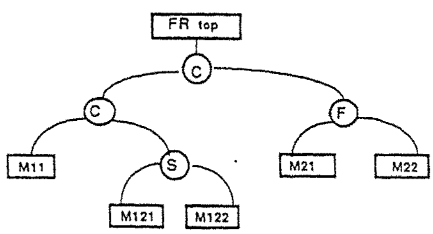


Fig.1 Module-junction structure diagram

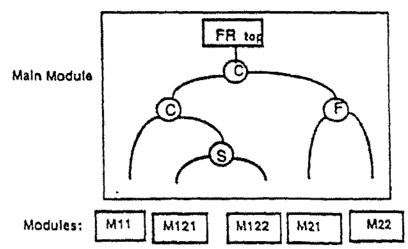


Fig.2 Modules and the main module

There are three types of junctions: summation(S), control(C) and feedback(F). When the child FRs are uncoupled, the parent FR is satisfied by combining all the outputs of its child modules. This is the summation junction(not an arithmetic summation). When they are decoupled, the parent FR will use the output of the left-hand side child module to control the execution of the right-hand side child module. This is the control junction. The coupled junction feeds back the output of the right-hand side modules to the left, which requires a number of iterations in processing data or in use of the program. When there are many sub-modules included in the feedback junction, the program will quickly become unmanageable. Obviously, the coupled junction must be avoided, especially during the decomposition process. By using the junction characteristics, the module-junction structure diagram can be easily converted to a network-type diagram which shows the flow of data stream among modules and has been the key graphical representation in conventional structured analysis and design methods.

Therefore, the result of Axiomatic Design is the starting point for the programmers who can code programs for each module, all of which then can be structured to develop the main module. This method facilitates both integration of the existing software and modification of software, since changes in one module does not affect other module and many functions that should not be affected

4. INTEGRATION OF DESIGN SOFTWARE FOR CONCURRENT ENGINEERING

Companies, no matter what kind of manufacturing enterprises they may be, always try to produce and offer the products the customers want, and then can make benefits for themselves. Therefore, according to Axiomatic Design, the first step is to know what the customers really require. In fact, the customers, no matter when and where they may be, always try to get the goods in time with better qualities and lower price, or the best qualities and the lowest price if possible, when they want it. It means that the general requirements of customers are all the same, which can be formalized in the functional domain as a set of functional requirements that govern how to integrate the different design method, design tools, design data and so on. Therefore, the functional requirements (FRs) at the first level can be listed as follows

FR_1 = The best functional qualities

FR_2 = The lowest material cost

FR_3 = Just in time

They are independent of each other. Thus, DP_1 should be selected to fulfil the FR_1; DP_2 should be selected to satisfy FR_2 but not to affect FR_1; and DP_3 should be selected to satisfy FR_3 but not to affect FR_1 and FR_2. An appropriate set of DPs that meet the FRs may be chosen as:

DP_1 = Using the most effective design methods

DP_2 = Choosing the substitute materials with the lowest cost

DP_3 = Shortening time to market

Therefore, the design equation can be obtained as follows:

$$\begin{Bmatrix} FR_1 \\ FR_2 \\ FR_3 \end{Bmatrix} = \begin{bmatrix} X & 0 & 0 \\ X & X & 0 \\ X & X & X \end{bmatrix} \begin{Bmatrix} DP_1 \\ DP_2 \\ DP_3 \end{Bmatrix} \quad (4)$$

where x represents a non-zero element and o represents zero element. From the equation obtained, it can be seen that the design matrix at this level is a triangular matrix which indicates that the design is a decoupled design and satisfies Independence Axiom. Therefore, the DPs can be adopted and the module-junction structure diagram of the system can be expressed as Fig.3. Since the FRs and DPs at the first level are not detailed enough, the further decomposition of the FRs is required.

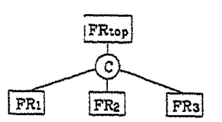


Fig.3 Module-junction structure diagram of software at the first level

4.1 FR_1

FR_1 is the best functional qualities and can be decomposed as:

FR_{11} = The best concept for the advanced performance

FR_{12} = The best appearance

FR_{13} = The best parameters for the advanced performance

FR_{14} = The best tolerances for the advanced performance

In order to satisfy the FRs at the second level, the effective DPs are now selected as:

DP_{11} = Using Conceptual Design or Creative Design

DP_{12} = Using Industry Design

DP_{13} = Using suitable parameter design methods

DP_{14} = Using suitable tolerance design methods

The design equation can be then obtained as follows:

$$\begin{Bmatrix} FR_{11} \\ FR_{12} \\ FR_{13} \\ FR_{14} \end{Bmatrix} = \begin{bmatrix} X & 0 & 0 & 0 \\ X & X & 0 & 0 \\ X & X & X & 0 \\ X & X & X & X \end{bmatrix} \begin{Bmatrix} DP_{11} \\ DP_{12} \\ DP_{13} \\ DP_{14} \end{Bmatrix} \quad (5)$$

From the equation obtained, it can be seen that the design matrix at this level is also a triangular matrix which indicates that the design is a decoupled design and satisfies Independence Axiom. Therefore, the DPs can be adopted. Since there are the existing design methods for DP_{11} and DP_{12}, the modules of computer software can be programmed according to Conceptual Design and Industry Design, and named as M_{11} and M_{12}. Thus, FR_{11} and FR_{12} do not need to be decomposed further and reach the terminal nodes in the FR_1 tree. But, DP_{13} and DP_{14} are not enough to fulfill FR_{13} and FR_{14} and the further decomposition is required.

4.1.1 FR_{13}

FR_{13} is the best parameters for the best functional qualities and can be decomposed further as:

FR_{131} = The best parameters for optimization of performance

FR_{132} = The best parameters for robustness of performance

FR_{133} = The best parameters for reliability of performance

FR_{134} = The best parameters for ease of manufacture

FR_{135} = The best parameters for ease of assembly

To satisfy FRs at the third level, the effective DPs can be selected as:

DP_{131} = Using Optimization Design

DP_{132} = Using Parameter Design of Robust Design

DP_{133} = Using Reliability Design

DP_{134} = Using Design for Manufacturability

DP_{135} = Using Design for Assembly

Therefore, the design equation can be obtained as follows:

$$\begin{Bmatrix} FR_{131} \\ FR_{132} \\ FR_{133} \\ FR_{134} \\ FR_{135} \end{Bmatrix} = \begin{bmatrix} X & 0 & 0 & 0 & 0 \\ 0 & X & 0 & 0 & 0 \\ 0 & 0 & X & 0 & 0 \\ X & X & X & X & 0 \\ X & X & X & X & X \end{bmatrix} \begin{Bmatrix} DP_{131} \\ DP_{132} \\ DP_{133} \\ DP_{134} \\ DP_{135} \end{Bmatrix} \quad (6)$$

From the equation obtained, it can be seen that the design matrix at this level is a triangular matrix which consists of a diagonal matrix of the first three elements and last two elements. It satisfies Independence Axiom. Therefore, the DPs can be adopted.

All these design methods for DPs at this level have existed already and even the computer software for some of these methods have existed too. Even if there is not the existing software for some of the design methods, the proper software can be programmed or developed according to the principles of these design methods. They can be named as M_{131}, M_{132}, M_{133}, M_{134} and M_{135}.

4.1.2 FR_{14}

FR_{14} is the best tolerances for ensuring the best functional qualities and can be decomposed as:

FR_{141} = The best tolerance for robustness of performance

FR_{142} = The best tolerance for reliability of performance

FR_{143} = The best tolerance for ease of manufacture

FR_{144} = The best tolerance for ease of assembly

To satisfy these FRs at this level, the effective DPs can be selected as:

DP_{141} = Using Tolerance Design of Robust Design

DP_{142} = Using Tolerance Design of Reliability Design

DP_{143} = Using Tolerance Design for Manufacturability

DP_{144} = Using Tolerance Design for Assembly

Therefore, the design equation for it can be obtained as follows:

$$\begin{Bmatrix} FR_{141} \\ FR_{142} \\ FR_{143} \\ FR_{144} \end{Bmatrix} = \begin{bmatrix} X & 0 & 0 & 0 \\ 0 & X & 0 & 0 \\ X & X & X & 0 \\ X & X & X & X \end{bmatrix} \begin{Bmatrix} DP_{141} \\ DP_{142} \\ DP_{143} \\ DP_{144} \end{Bmatrix} \tag{7}$$

From the equation obtained, it can be seen that the design matrix at this level is a triangular which consists of a diagonal matrix of the first two elements and the last two elements. It satisfies Independence Axiom. Therefore, the DPs can be adopted. All these design methods for the DPs at this level have existed and can be programmed into the modules of computer software as M_{141}, M_{142}, M_{143} and M_{144}. Up to now, all branches of the FR_1 tree form the terminal nodes and the decomposition process terminates. The module-junction structure diagram of software can be now drawn as Fig. 4.

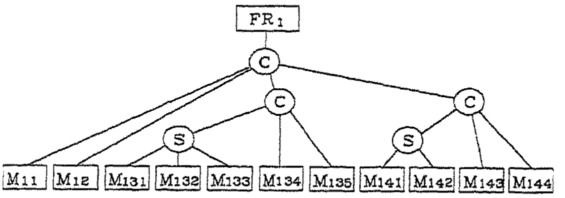


Fig.4 Module-junction structure diagram of the FR_1 tree

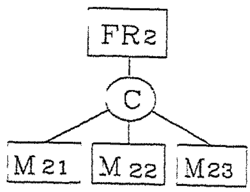


Fig.5 Module-junction structure Diagram of the FR_2 tree

4.2 FR_2

FR_2 is the lowest material cost and can be further decomposed as:

FR_{21} = Satisfying technological requirements

FR_{22} = The lowest price

FR_{23} = Shorter purchasing time

In order to satisfy FR_2 at the second level, the effective DPs are now selected as:

DP_{21} = Choosing from the list of all material properties

DP_{22} = Choosing from the list of all material prices

DP_{23} = Choosing from the list of purchasing times of all materials

Thus, the design equation for it can be obtained as follows:

$$\begin{Bmatrix} FR_{21} \\ FR_{22} \\ FR_{23} \end{Bmatrix} = \begin{bmatrix} X & 0 & 0 \\ X & X & 0 \\ X & X & X \end{bmatrix} \begin{Bmatrix} DP_{21} \\ DP_{22} \\ DP_{23} \end{Bmatrix} \tag{8}$$

From the equation obtained, the design matrix at this level is a triangular matrix which indicates that the design is a decoupled design and satisfies Independence Axiom. Therefore, the DPs can be adopted. All these DPs are searching the certain values in the certain database. The programs can be easily obtained and named as M_{21}, M_{22} and M_{23}. Thus FR_{21}, FR_{22} and FR_{23} do not need to be decomposed further and reach the terminal nodes in the FR_2 tree. The module-junction structure diagram of corresponding software for FR_2 can be drawn as Fig.5.

4.3 FR_3

FR_3 is just in time and can be decomposed as follows:

FR_{31} = The shortest design time of all processes
FR_{32} = The shortest time of fabrication
FR_{33} = The shortest time of assembly
FR_{34} = The shortest time of delivery of products

To satisfy FR_3 at the second level, the effective DPs are now selected as:

DP_{31} = Concurrent design by using computers
DP_{32} = Designing the best fabrication system to shorten the fabrication time
DP_{33} = Designing the best assembly system to shorten the assembly time
DP_{34} = Designing the best delivery system to shorten the delivery time

Thus, the design equation for it can be obtained as follows:

$$\begin{Bmatrix} FR_{31} \\ FR_{32} \\ FR_{33} \\ FR_{34} \end{Bmatrix} = \begin{bmatrix} X & 0 & 0 & 0 \\ 0 & X & 0 & 0 \\ 0 & 0 & X & 0 \\ 0 & 0 & 0 & X \end{bmatrix} \begin{Bmatrix} DP_{31} \\ DP_{32} \\ DP_{33} \\ DP_{34} \end{Bmatrix} \tag{9}$$

DP_{31} is what we are doing for and the integration of design software is for concurrent design by using computer. DP_{34} is designing the best delivery system to shorten the delivery time, which can be designed easily according to the company and the market, and the software for it can be named as M_{34}. DP_{32} and DP_{33} are very complicated and are not detailed enough to fulfill FR_{32} and FR_{33}. Therefore, the further decomposition is required.

4.3.1 FR_{32}

FR_{32} is the shortest time of fabrication and can be decomposed as:

FR_{321} = The least kinds of parts
FR_{322} = The most effective equipments
FR_{323} = The most effective tools
FR_{324} = The most effective delivery equipments
FR_{325} = The best quality control
FR_{326} = The best floor layout
FR_{327} = The lowest cost of fabrication system

In order to satisfy the FRs at this level, the effective DPs are selected as:

DP_{321} = Using Group Technology
DP_{322} = Using Draper's method (CTIME)[4]
DP_{323} = Using the effective design method of tools
DP_{324} = Using the effective design method of delivery equipment
DP_{325} = Using the effective design method of quality control
DP_{326} = Using MVAQ[5] and simulation
DP_{327} = Using the effective economic analysis method of systems

Thus, the design equation for it can be obtained as follows:

$$\begin{Bmatrix} FR_{321} \\ FR_{322} \\ FR_{323} \\ FR_{324} \\ FR_{325} \\ FR_{326} \\ FR_{327} \end{Bmatrix} = \begin{bmatrix} X & 0 & 0 & 0 & 0 & 0 & 0 \\ X & X & 0 & 0 & 0 & 0 & 0 \\ X & X & X & 0 & 0 & 0 & 0 \\ X & X & X & X & 0 & 0 & 0 \\ X & X & X & X & X & 0 & 0 \\ X & X & X & X & X & X & 0 \\ X & X & X & X & X & X & X \end{bmatrix} \begin{Bmatrix} DP_{321} \\ DP_{322} \\ DP_{323} \\ DP_{324} \\ DP_{325} \\ DP_{326} \\ DP_{327} \end{Bmatrix} \tag{10}$$

From the equation obtained, it can be seen that the design matrix at this level is a triangular matrix which satisfies Independence Axiom. Therefore, the DPs can be adopted. Since there are the existing methods for these DPs, the modules of computer software for them can be programmed and named as M_{321}, M_{322}, M_{323}, M_{324}, M_{325}, M_{326} and M_{327} respectively.

4.3.2 FR_{33}

FR_{33} is the shortest time of assembly and can be decomposed as follows:

FR_{331} = The best sequence of assembly
FR_{332} = The most effective assembly equipment
FR_{333} = The most effective delivery equipment
FR_{334} = The best quality control
FR_{335} = The best floor layout
FR_{336} = The lowest cost of assembly system

To satisfy the FRs at this level, the effective DPs are selected as:

DP_{331} = Using Boothroys' Method[6], De Fazio's Method[7] or Lui's Method[8]
DP_{332} = Using Draper's method[4]
DP_{333} = Using Suri's Method[5]
DP_{334} = Using Raz's Method[9]
DP_{335} = Using simulation
DP_{336} = Using the effective economic analysis method of systems

Thus the design equation can be obtained as follows:

$$\begin{Bmatrix} FR_{331} \\ FR_{332} \\ FR_{333} \\ FR_{334} \\ FR_{335} \\ FR_{336} \end{Bmatrix} = \begin{bmatrix} X & 0 & 0 & 0 & 0 & 0 \\ X & X & 0 & 0 & 0 & 0 \\ X & X & X & 0 & 0 & 0 \\ X & X & X & X & 0 & 0 \\ X & X & X & X & X & 0 \\ X & X & X & X & X & X \end{bmatrix} \begin{Bmatrix} DP_{331} \\ DP_{332} \\ DP_{333} \\ DP_{334} \\ DP_{335} \\ DP_{336} \end{Bmatrix} \quad (11)$$

From the equation obtained, it can be seen that the design matrix at this level is a triangular matrix which indicates that the design is a decoupled design and satisfies Independence Axiom. Therefore, the DPs can be adopted. Since there are the existing methods for these DPs, the modules of computer software for them can be programmed and named as M_{331} ,M_{332}, M_{333}, M_{334}, M_{335} and M_{336}. The module-junction structure diagram of corresponding software for FR_3 can be drawn as Fig.6.

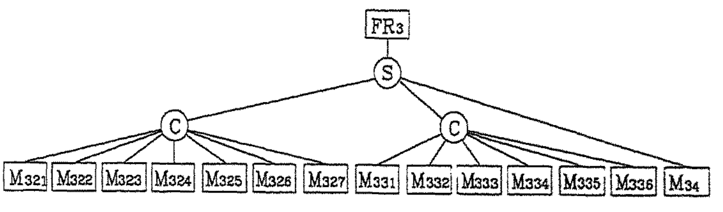


Fig.6 Module-junction structure diagram of software of the FR_3 tree

Thus, the whole module-junction structure diagram of computer software system for concurrent engineering can be obtained by integrating all the diagrams for FR_1, FR_2 and FR_3 as Fig.7.

5. CONCLUSION

Concurrent Engineering can help manufacturing enterprises to achieve shorter time to market, reduced development costs, and high quality products. In order to realize the concurrent engineering, it

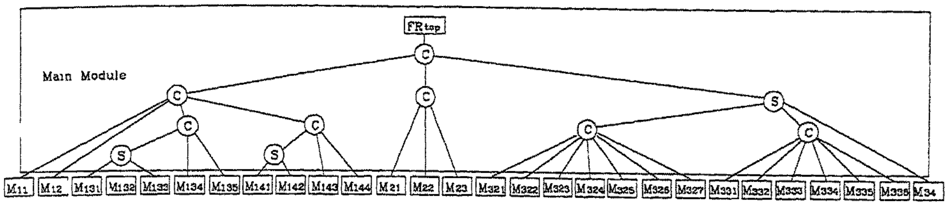


Fig.7 Module-junction structure diagram of the whole software system

is necessary to integrate effectively the people with different disciplines, design methods, design tools, design data, and so on. It means that the organizational structures must be changed and computers must be used with the integrated software system for concurrent engineering. There are many methods for integration of software system. But, to achieve the effective integration, the correct method must be used, which is Axiomatic Design. By using Axiomatic Design, a very complicated software system for concurrent engineering becomes very simple, and has 28 modules corresponding to 28 design methods and one main module, as shown in Fig.7, which contains all the junctional properties at each level The task of the programmer for the integration becomes very clear and is mainly programming for the main module.

ACKNOWLEDGMENT:

I would like to take this opportunity to express my sincere acknowledgment to Professor Nam P. Suh, Head of Department of Mechanical Engineering in Massachusetts Institute of Technology, U S.A. for his advice and assistance during the period of this work when I was visiting MIT.

REFERENCES:

1. Chen, K. Z., "Analysis of Relationship among Design Methods", Proceedings of International Conference on Manufacturing Automation, The University of Hong Kong, Hong Kong, 1997.
2. Suh, N. P., The Principles of Design, New York: Oxford University Press, 1990.
3. Kim, S-J, Suh, N. P., and Kim, S-G, "Design of Software Systems Based on Axiomatic Design", Robotics and Computer Integrated Manufacturing,Vol.8, No.4, pp. 243-255, 1991.
4. C. S. Draper Laboratory, "Decision Support Software Guides," in Flexible Manufacturing System Handbook, Vol. 5-A, C. S. Draper Laboratory Report R-1599, 1982(a), pp. 3-52. "Software Packages," in Flexible Manufacturing System Handbook, Vol. 5, C.S. Draper Laboratory Report R-1599, 1982(b), pp. 25-86.
   Users' Guide for the Cycle Time Calculation Software Package, rev. 1, December, 1984.
5. Suri, R., and R. R. Hildebrant, "Modeling Flexible Manufacturing Systems Using Mean Value Analysis," J. Manuf. Sys., Vol. 3, No. 1, 1984, pp. 27-38.
6. Boothroyd, G., "Assembly Automation and Product Design", Marcel Dekker, Inc., 1992.
7. DeFazio, T. L. and Whitney, D. E., "Simplified Generation of all Mechanical Assembly Sequences", IEEE J. Robotics and Automation, Vol. RA-3, No. 6, December, 1987, pp. 640-658. "Correction to 'Simplified Generation of All Mechanical Assembly Sequences'", IEEE J. Robotics and Automation, Vol. RA-4, No.6, December, 1988.
8. Lui, M. M., "Generation and Evaluation of Mechanical Assembly Sequences Using the Liaison-Sequence Method", MS Thesis, MIT Mech. Eng. Dept., May, 1988.
9. Raz, T., "A Survey of Models for Allocating Inspection effort in Multistage Production systems", J. Quality Technology, Vol. 18, No.4, October, 1986, pp. 239-247.
10. Nevins, J. L., Whitney, E. and DeFazio, T.L. et al., "Concurrent Design of Products and Processes: A Strategy for the Next Generation in Manufacturing", McGraw-Hill Inc., 1989.
11. Clausing, D., "Total Quality Development: A Step-by-Step Guide to World-Class Concurrent Engineering", New York: ASME Press, 1994.

PRODUCT DEVELOPMENT PROCESS SIMULATION IN CONCURRENT ENGINEERING

Keshu Zhang, Xiaochun Tan, JiaDong Chen
(CIMS-ERC, Tsinghua University,Beijing, P.R.China, 100084)

ABSTRACT

The characteristics of concurrent engineering determines that simulation plays an important role in it. The simulation for its product development process is especially beneficial in that it ensures the smooth implementation of concurrent engineering. In this paper, we give a detail analysis of the concurrent engineering product development process(CEPDP) and its requirements to simulation. An implementation schema of the CEPDP simulation tool is also suggested to fulfill some of these requirements.

KEYWORDS

Concurrent Engineering(CE), Product Development Process(PDP), Simulation

1. INTRODUCTION

Concurrent engineering aims at decreasing product lead time and cost while improving product quality. It focuses on the design phase of new product development. Questions such as manufacturability and assemblability should be considered as early as possible in order to avoid the time-consuming and costly product redesign. Simulation as a powerful analyzing tool is widely used in the area of manufacturing. Although a great number of simulation software were developed to help product managers in the planning, scheduling and control of their plant, most of them are focused on the design of product and manufacturing process(e.g. DFA, DFM, MPS). And they turned out not to be efficient enough to deal with the kind of questions encountered in the product development process. As product development process integrates management, resources, organization and information together, its design is a key factor that affects the implementation of concurrent engineering. Thus, efficient simulation tools are required to assist in the design and decision making of the product development process.

In the following sections of this paper, we will first investigate the features of product development process in detail, then the requirements it makes to simulation tools. Finally, we present a schema of implementing the simulation for product development process on the computer.

2. CONCURRENT ENGINEERING PRODUCT DEVELOPMENT PROCESS(CEPDP)

Here, by "process", we mean a set of activities which, executed in fixed successions with given different types of specific input, increase the added value at each single step and creates an output regarded as valuable by the customer[1].

In the traditional product development pattern, these activities are carried out in a sequential order(shown in fig. 1). The different functional departments work only on the processes that directly relevant to themselves: for example, product designers only consider the performance of the product, paying little attention to the process design and manufacturing capability. Little information exchange exists between the upstream and downstream design processes. This sometimes causes remarkable prolongation of the lead time of a product.

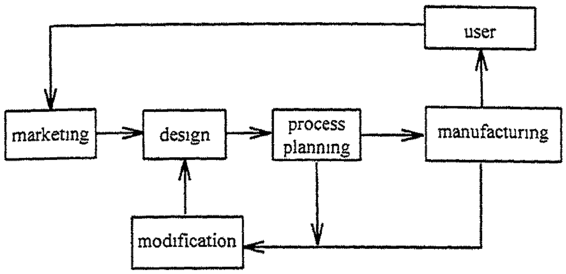


Fig. 1 Traditional Product Development Process

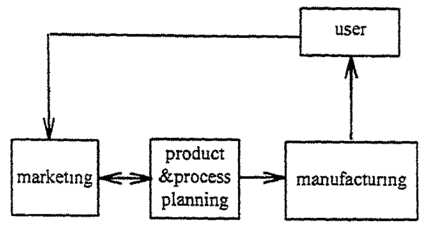


Fig. 2 CEPDP

But in concurrent engineering, things are different. Multi-discipline teams are formed to consider as much as possible factors that are related to the product life cycle, quality, cost, maintenance, etc., in the design stage. As there are much information exchange between processes, there is no explicit boundaries between them. The processes are carried out concurrently(shown in fig. 2). This helps to discover the flaw of the product design as early as possible.

Based on our investigation to many of the factories implementing concurrent engineering, we find that CEPDP has many characteristics that are different from the traditional PDP. They are listed as follows.

- CEPDP is a dynamic and variable developing task flow, which is supported by computer hardware, software and network environment. CEPDP emphasizes the communication between its participants. It is based on the information integration. People involved in it should own the whole and latest information relevant. This requires a powerful communication infrastructure.
- CEPDP is the intermedium that brings the information flow, organization and resources of a design project together. It involves a lot of objects, including various kinds of physical entities such as resources and designers, and virtual information entities such as information flow and activities.
- The boundaries between processes were broken. Multi-discipline team consists of product designers and persons from pertinent fields is adopted in concurrent engineering. This team works on all the processes in the product life cycle. As many information are considered at the same time, there are no definite start and end marks of a process.
- Human factor takes an important role in CEPDP. Concurrent engineering mainly changed the management pattern of a factory. It raises higher requirement to its participants. So the performance of the participants have a great impact on the implementing of CEPDP.

- There are many stochastic factors in CEPDP. These factors include task process time, unexpected event that affect the proceeding of a project(e.g. the absence of a participant of a review process), and the possibility of a design fault.

Due to these differences, analyzing tools that are suitable for the traditional design process(such as project planning, workflow, IDEF method) can not satisfy the new requirement of CEPDP any more. Thus the development of new analyzing tools for CEPDP are needed. Since simulation has proved to be a powerful analyzing tool in the area of discrete event systems, it is naturally adopted to assist in the design of CEPDP.

3 SIMULATION FOR PRODUCT DEVELOPMENT PROCESS IN CE

Simulation is assumed to tackle various issues during the evolution of CEPDP. These issues are

- How to simulate the frequent information exchanges between processes.
- Determination of the duration of product development. Investigate whether the deadline of the project can be met, while guarantee the product quality and development costs.
- Determination of resource and personnel utilization, illumination of complex interaction in CEPDP.
- Determination of how to deal with various stochastic exception which may occur during the product developing.

With the development of simulation technology, the object-oriented simulation provides a practical approach to simulation modeling and to simulation implementation. There are many reasons why object-oriented simulation have substantial benefits to CEPDP modeling[2].

The object-oriented system provide a design philosophy. Objects provide both data abstraction and information hiding that help to modularize a problem. Inheritance permits model to be reused and new objects to be added to existing models by using function and operator overloading. moreover, there is a substantial reduction in the size of the resulting code. This is more important to analyze product development process.

In object-oriented simulation, objects represent physical entities of the real world. Generally they can be represented pictorially. Therefore, object-oriented simulation models often have a natural pictorial (iconic) representation and are easily animated. Usually we can easily translate this simulation model into an animated simulation without additional conceptual changes.

Objects contain their own functionality, so intelligence can be built directly into this functionality using the machinery of artificial intelligence and expert systems. Further, this "intelligence" may be updated through simulated experiences as well as relying on various condition-action rules. Eventually objects would be designed that could "explain" simulation behavior as long as term information is gathered.

Objects provide a natural basis for concurrent. It means that each object could be assigned to its own processor and work away until it needed some form of coordination. Although it isn't clear exactly what form the coordination should take, There is a natural division among the simulation components when viewed as objects.

The current study has addressed the utility and desirability of using the object-oriented paradigm as the basis for product development process modeling in a highly dynamic and competitive environment.

3.1 The problem in product development process simulation

Simulation should evaluate various dynamic and stochastic events emerging during the implementation of CEPDP. There are some problems which are closely bound up with the implementation of simulation. These problems are listed as follows:

1. How to describe CEPDP and its intermedium actions that brings the information flow, organization and resources of a design project together.
2. How to deal with the complexity of CEPDP model. Multi-level model may facilitate CEPDP modeling, then how to confirm the level boundaries.
3. How to formalize the descriptions of CEPDP performance, covering evolution time, costs and quality? The formalization will assist quantitative analysis afterwards.
4. How to use various resources(i.e. human resource and data resource, etc.), How to use history or current knowledge and data, which will facilitate our product development?
5. How to deal with various exception during the project implementation? For example, how to handle the current tasks if an urgent mission come; How to arrange the work again if there are some personnel changes, etc..
6. How to adapt to the requirements of swift and simple simulation?

Answering Question 1 and 2 provides an integrated and agile CEPDP model which will illustrate wide relations between processes and various other physical entities, and facilitate simulation in various level(macro or micro) . Answering Question 3 provides an instrument to judge the performance of CEPDP instances. Answering Question 4 indicates the CEPDP implementing mode, which should be built in developing environment. Answering Question 5 provides robust capability for CEPDP. Answering Question 6 indicates some requirements to modeling and simulation tools.

Next section will consider these questions as general motivation for defining CEPDP model and CEPDP simulation.

3.2 Modeling Product Development Process in CE

The goal of CEPDP modeling is to create simulation models that answer the questions brought forward by multiple corporate functions (e.g., design engineering, manufacturing, business planning). The ultimate aim in the modeling is the design and specification of single model which provides information on CEPDP performance as well as financial and costing information, quality, and support system performance(e.g., maintenance). Modeling could be used to develop realistic information on the cost and performance impact of product schedule changes, product mix changes, or possible situation in which CEPDP simulation may implemented.

The characteristics of CEPDP determine that CEPDP model should build wide relations between various physical entities. Figure 3 is the base object class model of CEPDP drawn out from our practice. All the classes defined in this hierarchy are abstract classes. An abstract class has no instances or objects, but is defined to enable the inheritance of common characteristics to subclasses.

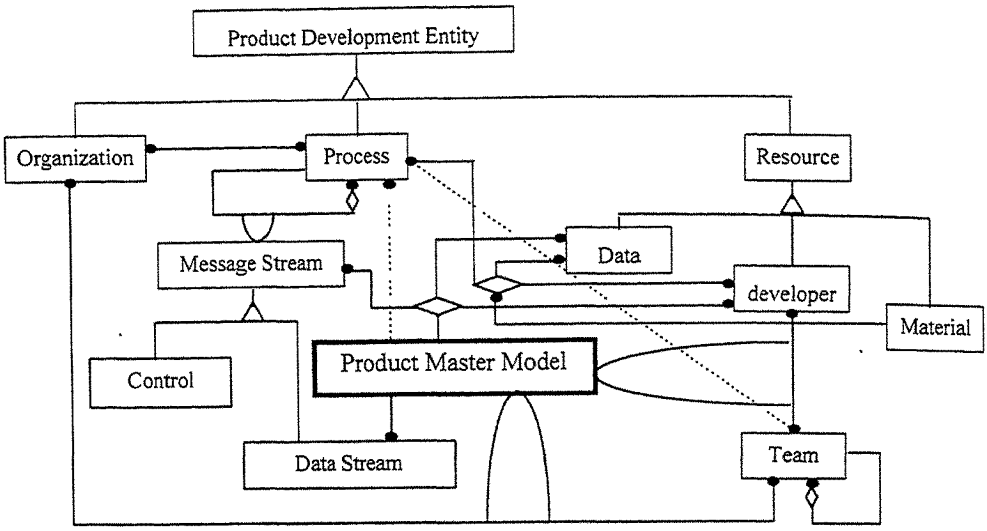


Figure 3 The CEPDP Simulation Model

In Figure 3, the process is defined as a time ordered series of activities that accomplishes a purpose. The ordered series of process activities are represented by recursive relationship between processes. This relationship is labeled Massage Stream. The recursive relationship between processes show that CEPDP model is a multi-level model. A complex project, especially if it is also large, is very difficult for a single person or even a small group to monitor and track the project's status information. So that there will have hundreds of work units(teams) in its work break down structure. As to CEPDP model, it requires the development of multi-level simulation.

The organization class is a mirror of enterprise function department. It represents element that impose maintenance and constraints to product development process and team activities[3]. Representing an organization as an object in relationship with other objects provides a more natural representation of the connection between project teams and the activities they perform. Specific behaviors, characteristics and relationships for the organization class are divided by the maintenance/constraints role that its play[4].

The Resource class is the enabler of process implementation. It is the super-class of Developer, Information and Material, which refine the exact roles played by the resource. Developer is the master of process implementation, it is the member of class Team. The relationship between Developer and Team is labeled product Master Model. The class Data represents various enterprise data which has relation with product development.

The class Team is the group of developers who have tasks in developing certain product[3].

Product Master Model(PMM) is the kernel of the whole process implementation. PMM is the parameteric model of product. It has wide relationship with Team, organization, Data Stream and Process own.

4. CONCLUSION

Simulation increases the understanding of product development process. Moreover, the product development process simulation is not isolated from enterprise environment. There are two aspects to further developing the product development process simulation further.

Simulation Environment: Simulation tool will be embedded into the whole environment of project process administration and management. In this environment, simulation tool can collect large amount of real time data as well as some history data from various information sources(i.e., MRP, MIS and PDM system). In this manner, simulation tool can analyze various cases to realize real time simulation, and use the results to direct production control and management.

Simulation Application: CEPDP simulation model is an integrated enterprise model. It can be used in the whole enterprise planning and management. Simulation tool of CEPDP can also be used as an assistant tool in Project Management(PM) as well as the supporting Business Process Engineering (BPR) in enterprise.

5. REFERENCE

1. Briccarello P., Bruno G., Ronco E., "REBUS: an object-oriented simulator for business processes", Proceedings of the IEEE Annual Simulation Symposium 1995, IEEE, Los Alamitos, CA, USA, pp 269-277, 1995
2. Stephen D. Roberts, Joe Heim, "A Perspective on Object-Oriented Simulation", 1988 Winter Simulation Conference, M. Abrams, P. Hailgh, and J. Comfort(eds), pp277-281, 1988.
3. Yan Jin, Raymond E. Levitt, John C. Kunz, Tore R. Christiansen, "The Virtual Design Team: A Computer Simulation Framework for Studying Organizational Aspects of Concurrent Design", SIMULATION, Vol. 64, pp160-174, March, 1995.
4. Barnett, W. D., Presely, A., Johnson, M., and Liles, D. H., "An Architecture for the Virtual Enterprise", 1994 IEEE International Conference on Systems, Man, and Cybernetics, pp509-519, IEEE Systems, Man, and Cybernetics Society, Texas, 1994.
5. Arun Jayaraman, Arun Agarwal, "Multi-Level Simulation Modeling in Engine Plants", AutoFact — Rapid Design & Manufacturing, pp158-168, Detroit, Michigan, 1996.
6. Phillip A. Farrington, James J. Swain, John L. Evans, and John S. Rogers, "Developing Reusavle Modeling Capabilities for Simulating High Volume Electronics Manufacturing Sustems", IEEE Transactions on Components, Packaging, And Manufacturing Technology - Part C, Vol. 19, NO. 2, April 1996

SURFACE MODELLING FROM CT-IMAGES

Weiyin Ma†, Bart Swaelens‡ and Wilfried Vancraen‡
†Department ME&EM, City University of Hong Kong, Hong Kong
‡Materialise N.V., Heverlee, Belgium

ABSTRACT

This paper summarizes techniques for constructing a CAD surface model from medical CT-images. The process starts from image segmentation and contour extraction. Image segmentation separates the areas of interests of a CT-image and identifies the corresponding boundaries. Contour extraction transforms the identified 2-dimensional boundary information into contours in 3-dimensional space. The extracted contours are further subdivided into individual topological surface areas represented by a set of base surfaces. The base surfaces are approximations of the final fitted surfaces and are used to parametrize the CT data for surface fitting. The techniques presented in this paper can also be applied for surface modelling from MR and industrial CT-images.

KEYWORDS

Computed Tomography, CT-images, B-Spline Surfaces, CAD

1. INTRODUCTION

CT stands for computer, computerised or computed tomography. CT-images have been widely used for medical diagnosis for more than 20 years. Combined with modern computer technology and manufacturing techniques, it is also possible to produce 3D medical models of anatomic or organic structures of a human body from the images of CT-scanners. Such 3D medical models serve as a hard copy for visualisation and they can be used to facilitate communication between surgical team members, between radiologists and surgeons, and more generally between medical doctors and patients. They can also be used for complex surgical planning and simulation before real surgical operations.

There are basically two types of 3D medical models, i.e., a tactile physical model and a computer model. The construction of these medical models from CT-images starts from segmentation and contour extraction. The extracted CT-contours can then be directly interfaced with rapid prototyping systems, such as stereolithography machines, for producing a physical part [1, 2]. The CT-contours can also be further processed to construct a 3D computer or mathematical model. Commonly used representations are triangular and polygonal approaches, i.e., the surface of a medical model is approximated by a triangular or a polygonal mesh. There are also systems that simply interpolate the coordinates and deflection angles of an array of points from the contours for visualisation.

While these 3D models are helpful for both visualisation and simulation, there are however difficulties in using the computer model for implant design and manufacturing. Most of the standard CAD/CAM systems do not support the above mentioned mathematical representations. Interfacing the above mentioned computer models with standard CAD/CAM systems remains a bottleneck for prosthesis design. As a result, most people still simply imports the CT-contours into a CAD/CAM system and uses the tools of the CAD/CAM system to construct a CAD model. This approach has two major problems. In order to create an accurate CAD model, one may need a lot of contours. As the contours are often represented by discrete points or poly-lines, there are a large number of points to be processed and most of the CAD systems are unable to handle such a large number of entities. In addition to this problem, the model construction process is manual and it takes a lot of time for constructing the CAD model.

This paper presents alternatives for constructing a digital 3D medical model suited for implant design. Instead of the triangular or polygonal mesh approaches, a B-spline surface model is constructed from the CT-images. Similar to other approaches, the procedure begins with the segmentation of the images and the surface contours are extracted from the CT-data. The extracted contours are further subdivided into areas representing different features of the model to be constructed. Various algorithms are then applied to create the individual surfaces from the CT-contours. The created surface model can then be exported in IGES format and the IGES file can be further imported into any standard CAD/CAM systems that support IGES.

2. A BRIEF OVERVIEW OF EXISTING APPROACHES

The raw data coming from a CT-scanner are a stack of two-dimensional arrays of grey values, often called slices of CT-images. Each of the slices corresponds to the density distribution of an intersection of a human body under investigation. Since the density of bone structures and soft tissue is different, one can clearly recognise internal bone structures from a CT-image. The objective is then to create a three-dimensional CAD model from these 2-D cross-sectional grey value images.

In literature, there have been various approaches for constructing a CAD model from the CT data. Most of these approaches use standard CAD/CAM systems as a key component. Blomer [3] and Khoo et. al. [4], for example, employed a CAD/CAM system called CATIA from Dassault Systems. Riek and Roth [5] used Unigraphics of EDS instead. Theoretically, one may use any CAD/CAM systems. Major steps of these approaches are the following:

- During the first step, image processing systems are applied to the CT-images for the separation of areas of interests and furthermore, extracting the contours from the boundary information. The extraction of contours can sometimes be done by using standard image processing software [5]. There are also specialised CT-image processing systems that provide contours with enhanced sub-pixel resolution [1, 6].

- The extracted contour information from CT-images is transferred to a standard CAD/CAM system in 3-D for further processing. In most of the cases, the contour information is represented by a large number of discrete points or a set of poly-lines. These contours must be further processed into cross sectional curves with fewer parameters in order to use these CAD/CAM systems for CAD surface or solid modelling. The creation of such cross sectional curves is also done by using the CAD/CAM systems themselves.

- When the cross-sectional curves are available, one can then use the CAD/CAM system to construct a CAD model from these cross sectional curves. In most of the cases, one creates a CAD surface model. A solid model can be created from the surface model when it is necessary. There are also approaches that directly create a solid model for rapid prototyping by extruding the contours by a thickness equivalent to the CT-scan slice spacing [5].

The above mentioned approaches solve some of the problems. There are however limitations in connection to the use of standard CAD/CAM systems. First of all currently available CAD/CAM systems are unable to handle a large number of entities. The number of slices from a CT-scanner for constructing a CAD model is often several tens or several hundreds. Each slice may have many contours depending on the complexity of the structure being investigated. Each of the contours may also contain several hundreds or thousands of discrete points. If one directly imports such a large number of entities into a CAD system, the system may simply be busy for displaying the entities and there would be no opportunity for a user to interest with the CAD system. To avoid the problem, most of the users split the CT data into batches and process a few tens of contours in a CAD session [3]. The second limitation of these approaches is the parametrization. In order to create an accurate CAD model, one often need to use a large number of contours. When creating surfaces from a large number of contours, however, one has to synchronise the parametrization of all the contours for creating a single surface. In other words, the parametrization of the individual contours must be compatible to each other. Otherwise, the created surface will exhibit irregular surface paths and may not be smooth. To avoid this problem, some users use only a limited number of contours for creating a single surface. This may however increase the total number of surfaces for representing the object under consideration. It also poses difficulties for smoothly connecting the surfaces. In addition to these problems, the above mentioned approaches involve a lot of manual processing.

During the past decade, there have been extensive research and development on reverse engineering that can be potentially applied for creating a CAD model from CT-data. Reverse Engineering is a process for creating a CAD model from a physical part through dimensional measurement [7]. The approach presented in this paper is actually a reverse engineering approach applied to medical applications.

3. THE PROPOSED APPROACH FOR SURFACE MODELLING

This paper presents a four-step approach for CAD surface modelling from CT-images, i.e., (a) CT image segmentation and contour extraction; (b) topological subdivision, (c) direct surface fitting and (d) CAD further processing. B-spline surfaces, and sometimes NURBS surfaces, are utilised for modelling the bone structures.

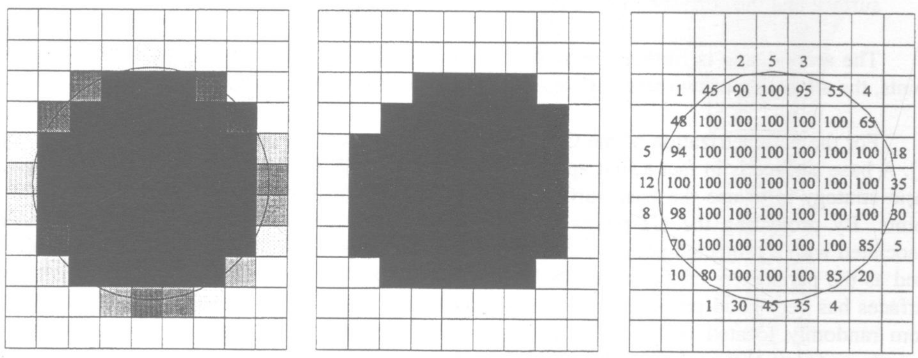
- The first step for constructing a CAD model from CT-images is to interpret each of the individual CT-images and determines which parts of a CT-image are bones and which parts are not. This process is often called image segmentation [1]. Contour extraction then determines the boundaries of the area of interests and extracts the boundaries as contours for further processing. This step is the same as other approaches summarised in the previous section and a CT-Modeller system from Materialise N.V., Belgium, is employed to perform the task.

- During the second step, the extracted contours are subdivided into surface areas. A collection of these surface areas represents the topological structure of the CAD model of the bone structures under investigation. Each of the individual surface areas is represented by a B-spline surface that is created from two or more contour curves, four boundary curves, or a combination of a set of contour curves plus the boundaries depending on the complexity of the surface area. These surfaces are called *Base Surfaces* and need not to be very accurate. They are used to parametrize the contour points in the following step for B-spline or NURBS surface fitting. The number and topology of the final fitted surfaces will be the same as that of the base surfaces.

- When the object is topologically subdivided into regions and all base surfaces are available, one can then perform direct surface fitting. For each of the surface areas, the fitting process starts from the parametrization of the contour points falling inside that area using the corresponding base surface. Two sets of knots are further allocated. The control points and the corresponding weights in case of NURBS are further identified using a standard least squares fitting process. Major steps for fitting surfaces using a base surface were reported in [8] and are also briefly summarised in the following section of this paper for easy reference.

- After direct surface fitting, one should have all the individual surfaces and these surfaces are further processed using available geometric processing tools for smooth connection. This step can usually be done by most of the standard CAD/CAM systems and will not be further elaborated in the present paper.

A commonly used approach for segmentation is the so-called "thresholding". All areas whose pixel grey value is higher than a certain value, i.e. the threshold, are assumed to be bones and the remaining areas are not. The same principle applies if we want to separate areas between cortical bones and soft tissue or between soft tissues and air. Since a CT-image has a fixed resolution in terms of the number of pixels, the boundaries determined by thresholding are usually not smooth and a stair effect would be obvious when the image resolution is low. This is a common problem since the resolution of most of the medical CT-scanners is between a square of 256×256 pixels and 512×512 pixels. To solve the problem, the CT-Modeller system uses a grey value interpolation between the pixels to obtain accurate and smooth boundaries from the CT-images [1]. Figure 1 provides a comparison between the segmentation results of simple thresholding and the results of thresholding plus grey value interpolation. In Figure 1 (a) one can find a simulated CT-image with the corresponding theoretical object boundary. Figure 1 (b) illustrates the segmented object using a simple threshold of 50 and Figure 1 (c) shows the grey value interpolated contour of the object. The contour obtained using grey value interpolation in Figure 1 (c) is far more superior than the contour obtained by using a simple thresholding in Figure 1 (b). Figure 2 illustrates a real CT-image and the corresponding CT-contours extracted from the CT-image using the CT-Modeller system.

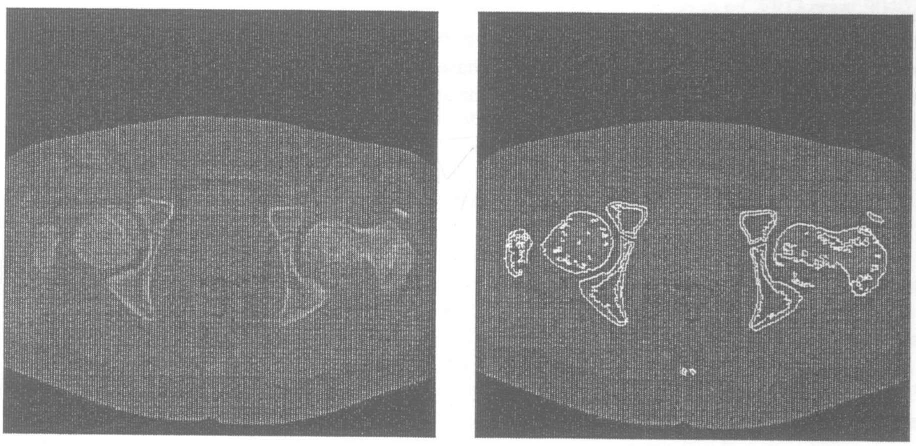


(a). Theoretical contour and computed CT-image

(b). Constructed object without interpolation (threshold 50)

(c). Interpolated contour and pixel value map

Figure 1. Comparison of simple thresholding and thresholding with pixel value interpolation.



(a). A CT-image representing a cross section of the hip

(b). The extracted contours of the hip bones from the CT-image

Figure 2. A real CT-image and the extracted contours

When segmentation and contour extraction are complete, the CT-contours are further subdivided into topological regions and the corresponding base surfaces are further created for further processing. At this moment, the subdivision is however still done manually and it demands a lot on the user's experience for CAD processing.

4. SURFACE MODELLING FROM CT-POINTS USING BASE SURFACES

When the base surfaces are available, we are then ready to perform direct surface fitting. The basic steps are the following:

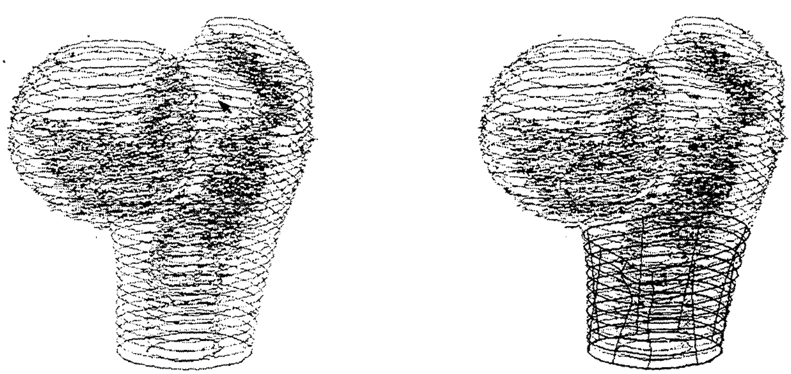
- All the CT-points are first partitioned into individual regions each of them corresponds to one of the afore-constructed base surfaces.

- For each of the individual regions, we can then perform direct surface fitting using the base surface and the corresponding CT-points.

The second step is further subdivided into three sub-steps, i.e., the parametrization of the CT-points, the allocation of two sets of knots and B-spline or NURBS surface fitting.

For the parametrization of the CT-points, the base surface parametrization technique is applied [8]. A base surface is in fact a first approximation of the final fitted surface of that surface area. The basic strategy is to use the base surface as a reference surface for the parametrization of the CT-points. By projecting the CT-points onto the base surface, each of the CT-points will then be associated with a projected point on the base surface. The parameters of the projected points are then used as the parameters of the corresponding CT-points for further fitting. Parametrization with base surfaces has very good properties. The most important property is the possibility for fitting surfaces from randomly located points. In addition, the use of base surface parametrization obtains very smooth surfaces with excellent parametrization compared with direct surface creation from CT-contours.

For the allocation of the knots, an average method is applied [8]. The average method tries to allocate more knots at places where there are denser CT-points than other places. This guarantees a very good conditioned observation system and consequently obtains near optimal knots for surface fitting. When the CT-points are parametrized and the knots are selected, one can apply standard least squares fitting techniques to identify the control points [10]. One can also find more information regarding the fitting of NURBS surfaces with given parametrization and knots in [9]. Figure 3 illustrates a set of CT-contours and one of the fitted surfaces using the base surface approach. Figure 4 shows a set of surfaces fitted from another set of CT-contours [7] using the approach summarised in the present paper.



(a). 3D contours of a piece of fumur extracted from the CT images

(b). The extracted contours and one of the fitted fumur surfaces

Figure 3. The constructed 3D contours and a created CAD surface

5. CONCLUSIONS

This paper presents the basic strategies and steps for constructing a CAD surface model from CT-images. The process starts from image segmentation and contour extraction. The object of interests is further subdivided into topological areas represented by a set of base surfaces. Numerical algorithms are applied to each of the areas for creating surfaces from the associated CT-contours using the corresponding base surfaces. The reported techniques also apply for creating a CAD model from MR (Magnetic Resonance) and industrial CT-images.

Future work includes the automation of some of the steps employed in the current approach. The topological issues in medical CAD modelling are also interesting. Comparatively it is more difficult to tackle the topology problem using rectangular B-spline patches than using the triangular or polygonal surfaces. The later surfaces are however not compatible with existing CAD/CAM system.

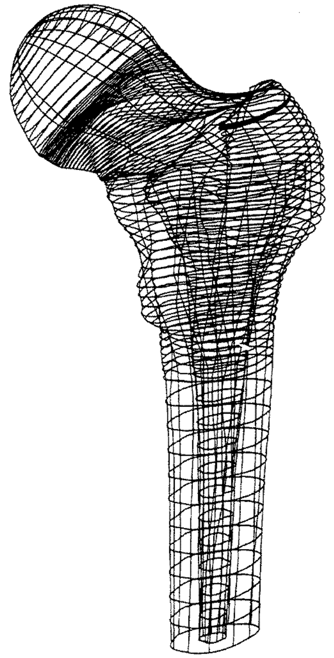


Figure 4. Some CAD surfaces created from a set of fumur contours

6. ACKNOWLEDGEMENTS

The algorithms presented in this paper are partially implemented in a software environment of Materialise N.V. in collaboration with Katholieke Universiteit Leuven, Belgium. The present research of W. Ma is sponsored by the University Grants Council of Hong Kong through a CERG grant 9040225.

REFERENCES

1. B. Swaelens and J.P. Kruth, "Medical Applications of Rapid Prototyping Techniques", Proc. of the 4th Int. Conf. on Rapid Prototyping, pp. 107-120, Dayton, OH, June 14-17, 1993.
2. J. Milberg; G. Reinsert; J. Moctezuma; "An Integrated Software Package for medical Stereolithography Based on Computer Tomography", Production Engineering, Vol. III, Nr. 1, pp.119-122, 1996.
3. W. Blömer; "From X-ray pictures to implants", CAENEWS, Nr. 3, pp. 13-17, 1994.
4. L.P. Khoo; J.C.H. Goh; S.L. Chow; "Parametric modeling of a knee joint prosthesis", Journal of Engineering in Medicine, Vol. 207, Iss. H2, pp.115-120, 1993.
5. A.T. Riek and R.N. Roth, "Stereolithography Models to Aid Revision Hip Replacement Surgery", Proc. of the 4th Int. Conf. on Rapid Prototyping, pp. 189-200, Dayton, OH, June 14-17, 1993.
6. J.H. Stanley, R.N. Yancey, Q. Cao and N.J. Dusaussoy, "CT-Assisted Solid Free-Form Fabrication", Proc. of the 6th Int. Conf. on Rapid Prototyping, pp. 267-274, Dayton, Ohio, June 4-7, 1995.
7. J.P. Kruth and W. Ma, "CAD/CMM Integration for Reverse Engineering", Laser Assisted Net Shape Engineering, M. Geiger and F. Vollertsen (eds.), Meisenbach Bamberg, pp.705-714, 1994.
8. W. Ma and J.P. Kruth, "Parametrization of randomly measured points for least squares fitting of B-spline curves and surfaces", Computer Aided Design, Vol. 27, No. 9, pp. 663-675, 1995.
9. W. Ma and J.P. Kruth, "NURBS Curve and Surface Fitting and Interpolation", Mathematical Methods for Curves and Surfaces, M. Daehlen, T. Lyche and L.L. Schumaker, (eds), pp 315-322, Vanderbilt University Press, Tennessee, 1995.
10. C. de Boor, A Practical Guide to Splines, Springer, 1978.

A LASER PROBE DEVELOPMENT FOR 3D SURFACE MEASUREMENT*

Xiaoqi TANG Pingjiang WANG Jihong CHEN Ji ZHOU
School of Mechanical Science and Engineering
Huazhong University of Science and Technology, Hubei ,China
Lilong Cai
Department of Mechanical Engineering
The Hong Kong University of Science and Technology
Clear Water Bay ,Kowloon, HK

ABSTRACT

A new noncontact measurement system using a laser probe for 3D surface measurement is presented in this paper. The measurement principle of this probe is based on the modification of the traditional triangulation principle using laser and linear CCD. This probe only uses one set of CCD and signal processing circuit instead of three sets as in reference[5] but still has the properties of less shadow effect and linearity. In order to increase the measurement speed and eliminate the pixel lost phenomenon of the CCD, a specific circuit for signal processing which is independent of the computer system is used. Moreover an algorithm for computing the center position of light spot on CCD by its gray-scale weight center is introduced to increase the measurement accuracy of the probe. This probe has been successfully used in the automatic measurement of unknown surface on Huazhong Type I Machining Center.

KEYWORDS

3D surface measurement, Laser noncontact probe, Linear CCD

1. INTRODUCTION

There is an increasing demand for measurement of 3D(three dimensions) surface with higher speed and accuracy in the areas of measurement of complex sculptured surface, molds, rapid prototyping, CNC(Computer Numerical Control) machine, and so on. To meet these requirements many methodologies for measurement of 3D surface have been proposed. One of them is the well-known optical method based on the triangulation principle with laser and CCD devices[1-4]. Much work has been done by many researchers. However, because of the nonlinearity and shadow effect existed in this principle, its applications are limited. For these reasons Miyoshi, T and et al[5] proposed an approved scheme that could partially overcome these disadvantages. The principle is as following:

In Fig.1 a laser beam projects on the target surface through a lens. The scattered light from the surface is collected by the same lens and focused by another one. The main axis of the two lenses is coincident. The focused light is then imaged as a circle on an image plane as in Fig.2a. Theoretically the radius r of the light circle is proportional to the displacement between the target surface and the laser. In order to measure the radius of the light circle and overcome the shadow effect, Miyoshi, T et al[5] put three linear CCDs, which are separated into 120 degrees on the image plane shown in Fig.2a. In this way no matter which direction the target surface inclines, there is always at least one CCD that can receive the reflected light. Therefore the shadow effect can be greatly decreased.

Obviously, this type of installation is very complex with large volume and complicated software because it uses three sets of linear CCD, driver and signal processing circuit.

In this paper we propose a simplified probe based on the same principle as in Fig 1. The

*This work is supported by the National Nature Science Foundation Grant of China

difference of this probe from [5] is that we only use one linear CCD instead of three ones and place it on the image plane with the center of CCD coincided with the center of the light circle as in Fig.2b. Therefore the reflected light will be imaged on the two sides of the CCD. In this case the volume of the probe is greatly decreased and signal processing circuit is simplified so as the algorithm. In addition, the property of the linearity and low shadow effect can also be held in the cost of narrower range of measurement. In order to increase the measurement speed and illuminate the pixel lost phenomenon of the CCD we also developed a kind of specific circuit for signal processing which is independent of the computer system and controlled under its own logic unit. Moreover, to overcome the affect of the attenuation of light source, stray light from the circumstance and electronic noise, we use an optimal algorithm to obtain the light center imaged on CCD through calculating the distribution center of the light spot. With these specific circuit and the algorithm of the light center combined together the measurement speed and accuracy of the probe can be largely increased.

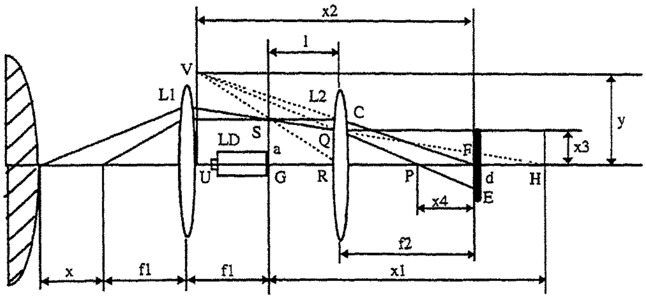
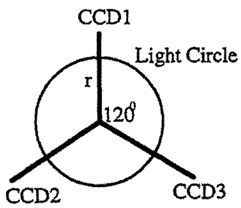


Fig. 1 Optical imaging principle of the probe



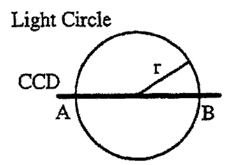


Fig.2a Three CCDs used

Fig.2b One CCD used

Fig.2 CCDs placed on image plane

2. MEASUREMENT PRINCIPLE OF THE PROBE

The optical imaging principle is shown in Fig.1. It consists of five main parts, which are semi-conductor laser LD(Laser Diode), lens L1, L2, diaphragm S and the receiver—high accurate linear CCD. The laser beam from LD is focused by lens L1 and then projected on the surface of the target object. The reflected light from the surface passes through diaphragm S and then imagined on the CCD by lens L2. From Fig.1 , we can see that after the transformation by the optical system the distance x between target surface and the focus of lens L1 is related to the light spot position d imagined on the CCD. Therefore the displacement of the target object x may be computed by measuring the position d of the light spot. According to the principle of geometry the relationship of d and x can be described by

$$d = \frac{a * f_2}{f_1^2} * x \qquad (1)$$

where f1, f2 are the focus of L1 and L2 respectively. a is the height of diaphragm S.The distance x between the surface of the target object and the focus of L1 is proportional to d, f2 and a , and reciprocal to the square of f1. As the structure of the optical system is fixed the value of a, f1 and f2 will remain constant. Therefore, as long as the radius d of the light circle on CCD is known the distance x can be computed by (1). From Fig.2b we can see that no matter which direction of the target object inclines there is always at least one of the two points A and B on CCD which can receive relatively much scattered light. So the probe is still able to have the less shadow effect. In this way only one set of CCD and signal processing circuit is needed in stead of three sets as in [5]. Therefore the system is largely simplified.

3. THE PROCESSING CIRCUIT OF SIGNAL

The methods of measurement with linear CCD is not the same as those of traditional ways such as inductors, capacitors and so on. Because for the former way only when all the pixels in CCD are read out then the testing result can be got according to the computed illuminated center on CCD. Therefore, for once measurement a certain time must be spent to read out all the pixels. This leads to a sensor-output bottleneck when high measurement speed is needed[6]. In general, the less the working frequency of linear CCD the much the time will be spent so as the less speed the measurement. For instance, a linear CCD TCD102D with 2048 pixels will spent about 20 ms to read out all the pixels with the working frequency of 100K Hz. This is too slow to be satisfied in the NC measuring system of real time. But for traditional way the measuring time is only the response time of electronics circuit and that of A/D conversion. It only needs a few micro-second.

In real time measurement system people wish the working frequency of CCD as high as possible in order to increase the measurement speed. But the higher working frequency is limited not only by the A/D conversion speed but also by the data sampling rate of the computer. Therefore, how to increase the speed of A/D conversion and data sampling is a key problem in linear CCD for increasing measurement speed. For instance, in a CCD with 2048 image pixels, reading out a pixel spends only several micro-second for the working frequency of more than 100K Hz. It is required that the computer and A/D conversion circuit must complete the analog to digit signal and then send the digital signal to a memory in such a short time. For simplicity, people often use high A/D conversion chip and direct memory access (DMA) to realize this purpose. But in practice, we discover even if for an advanced computer like PC386 its highest DMA transmission rate is less than 300K bytes per second. Moreover the transmission rate of DMA is influenced by the complexity of computer and control system. This is because as computer transmits CCD data to memory through DMA the synch pulse SH of CCD is read by scanning or interrupt. When SH appears, if just at this time, the CPU is running a prior interrupt program it will ignore SH or process it after the prior interrupt program is finished. Therefore, DMA will not probably response immediately when the first pixel of CCD behind SH appears. This is called pixels lost phenomena. That is, the sampled digital signal by CPU could not be consistent with the corresponding image pixels of the CCD completely. The higher the working frequency of CCD the more serious the pixels lost will be. This will greatly decrease the measurement accuracy of the probe. In order to solve pixels lost phenomena we designed a special A/D conversion and data sampling circuit as shown in Fig. 3. Its working principle is as following:

The time sequence generator produces the signals Φ_1, Φ_2, SP, RS and SH which are required by CCD of TCD102D[7]. The working frequency of CCD is 500KHz. The image pixel output analog signal of CCD is amplified and filtered by a low-pass filter and then converted to digital data through high speed A/D circuit which is controlled by the reset pulse SP[8]. The digital data are then stored in a static RAM by the following steps.

- The control logic unit opens the gate G1 waiting for the synch pulse SH;

- As the synch pulse SH appears it resets the address generator(output 0000H). Then SH closes G1 through control logic unit so as to prevent the SH afterwards;
- Control logic unit opens G2. A/D conversion ended pulse EO makes the address generator have an increment by G2. Meanwhile, EO opens the static RAM and the output digital data of A/D is written into static RAM consequently from the address 0000H under the control of address generator.

After all the digital data of the CCD pixels are stored in RAM the control logic unit closes G2 to blockade the Address generator. Then the computer reads out all the data from RAM for computing the center point of light spot on CCD. It follows the following steps.

- Control logic unit clears the address generator.
- CPU uses the instruction of inportb (I/O port) to generate a pulse for address increment as well as the reading pulse for RAM.
- CPU reads data from RAM consequently until all the data are read out.

Experiment shows that the measurement data of CCD transformed to computer using this special hardware can not only increase the measurement speed but also resolve the problem of pixels lost completely.

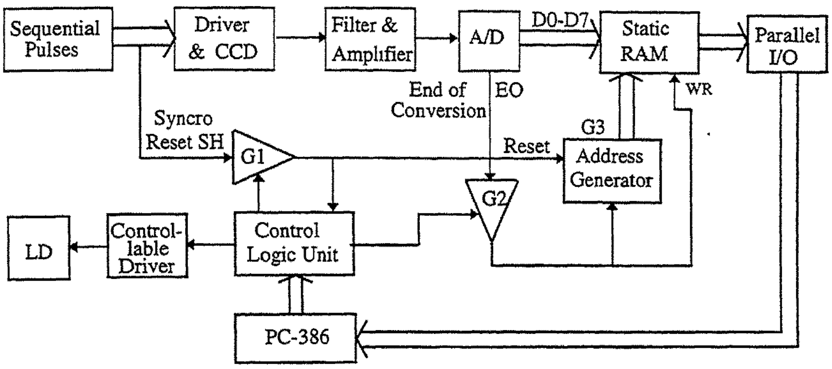


Fig. 3 A/D Schematic diagram of conversion and signal sample circuit

4. COMPUTATION OF CENTRAL POINT OF THE SPOT

In most cases, the shape of the illuminated spot on CCD is almost like an ellipse. Its size is related to its position on CCD surface. The nearer the spot is closed to the center the smaller its size will be. The numbers of illuminated pixels of the CCD are in the range of fifty to two hundred. Therefore how to compute the exact position of the central point of the spot(CPS) on CCD surface is very important to measuring accuracy of the probe.

Normally, as long as the illuminated pixels of CCD are not saturated the intensity profile of the spot is like a Gaussian type as shown in Fig.4. We have three ways to compute the CPS. First, CPS is the horizontal axis value related to the maximum value point of the curve. Second, it equals to the average value of the horizontal axis. Third, CPS is the numerical centroid of the light spot. Theoretically if the intensity profile is of ideal Gaussian profile the computation results will be the same for the three methods. However, in practice, the spot profile is not the same as an ideal Gaussian type, so there exists large difference in the above methods. Alexander and Ng[9] proved from simulation and experiment that the third way can eliminate the systematic error and improve the measurement accuracy. So we use the centroid way to compute the CPS.

In Fig.4 let most left side of CCD be the original point of the coordinate axis, its pixel position be X_k on the horizontal axis and their relative illuminated intensity as I_k on the vertical axis. The CPS X_0 can be described by:

$$X_0 = \sum_{k=0}^{2047} I_k X_k / \sum_{k=0}^{2047} I_k \tag{2}$$

If the intensity profile is near to symmetry the computed CPS by above is consistent with the maximum point of light intensity. However, in most cases the profile is not of symmetry so CPS will have some difference with the maximum point as shown in Fig.4a. In this case the linearity of the probe will be affected. Therefore we use a modified centroid way to compute CPS as in the following:

$$X_0 = \sum_{k=0}^{2047} I_k^n X_k / \sum_{k=0}^{2047} I_k^n \tag{3}$$

Where n is the weighting factor with the value of 1 to 5. From our experiment as long as n is selected suitably the linearity and stability of the probe may be improved. The consistence of the maximum point and CPS is shown in Fig.4b.

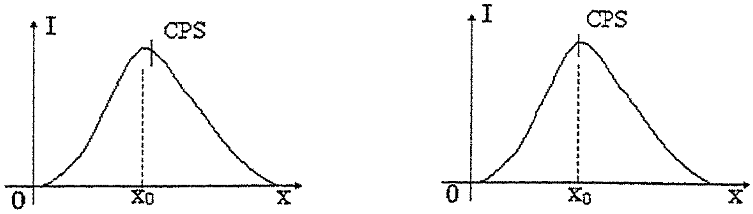


Fig. 4a CPS by (2) Fig. 4b CPS by (3)

Fig. 4 Intensity profile of the light spot on CCD

5. PROBE DESIGN AND EXPERIMENT

We have produced a laser probe prototype based on the principle of Fig.1 with the value of $f_1 = 45mm$, $f_2 = 90mm$, $a = 15mm$. The type of linear CCD is TCD102D with 2048 pixels. The width of each pixel is $14\mu m$, So its illuminated length is about $28mm$. The sensitivity of the probe is $d / x = (15*90)/(45*45) = 2/3$. The measurement range is $28mm*(1/2)*(2/3) \approx 9mm$ and the resolution is $k = 14\mu m/(d/x) = 21\mu m/unit$. Fig.5 is the structure diagram of the probe.

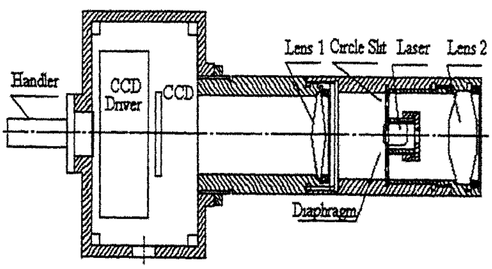


Fig. 5 Structure diagram of the probe

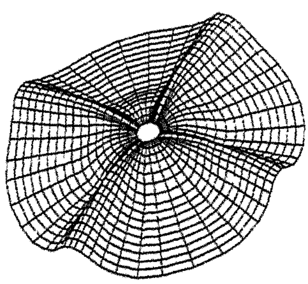


Fig. 6 Measurement result of a 3D surface

We use this probe combining with the measurement software to measure a 3D surface of a spindle propeller of a washing machine on Huazhong Type I NC machine center. The probe and the propeller are fixed on the vertical axis and the table of the machine respectively. So the probe can automatically move up and down by the NC servo system as the detected spot always exists within the measuring range of the probe. The movements of the horizontal axis of X and Y are under the control of the NC measurement software for scanning the surface of the object. Moreover the profile of a 3D surface of the object can be seen in real time as shown in Fig.6.

6. CONCLUSION

The prototype of the laser probe which we have produced is very simple in structure and easy for installation. The experiment shows that the special signal process circuit and the CPS of weighting factor algorithm can improve the speed and accuracy of the measurement for unknown surface with less shadow effect and a better accuracy. However in practice we discovered that the measuring accuracy will be influenced by the following factors. (a) manufacturing accuracy limit of the probe parts so the linearity of the probe is not absolutely the same as in (1). (b) The reflect coefficient change of the object target and intensity attenuation of light source result in the inconsistence of the spot profile on CCD. Experiment shows that this probe, with a measuring range of $8\,mm$, can measure the displacement of the free-form surface within an accuracy of $\pm 20\,\mu m$ for a steep inclined surface up to 60 degrees.

7. REFERENCES

1. Wiese, D.R., Laser triangulation sensors: a good choice for high rate inspection, I&CS pp.27-9,vol.62, no.9, 1989.
2. Milbocker, M.T., Reznichenko, Y.M., Comparison of three-dimensional retinal imaging methods: the method of scanning laser triangulation, Applied Optics, pp.4148-53, vol.30, no.28, 1991.
3. Dalglish, R.L. McGarrity, C. Restrepo, J., Hardware architecture for real-time laser range sensing by triangulation, Review of Scientific Instruments, pp.485- 91, vol.65, no.2, 1994.
4. Mayer, J.R.R. Parker, G.A., A portable instrument for 3-D dynamic robot measurements using triangulation and laser tracking, Transactions on Robotics and Automation, pp.504-16, vol.10, no.4, 1994.
5. Miyoshi, T. Kondo, T. Saito, K. Kamiya, Y. Okada, H., Development of noncontact 3-D digitizing system. Journal of the Japan Society of Precision Engineering, pp.1021-6, Vol.56, no.6, 1990.
6. Harrison,D.D.Weir, M.P., High-speed triangulation-based 3-D imaging with orthonormal data projections and error detection, IEEE Transactions on Pattern Analysis and Machine Intelligence, pp.409-16, Vol.12, no.4, 1990.
7. Cai Wengui, The applications of CCD technology, Electronics industrious press, Beijing, 1992.
8. Xiaoqi T., Pingjiang W., A kind of A/D conversion circuit with high speed and low cost, Electrical measurement and instrument, pp.40-42, No.2, 1995.
9. B. F. Alexander and K. C. Ng, Elimination of systematic error in subpixel accuracy centroid estimation, Opt. Eng. pp.1320-1331, Vol.30, No. 9, 1991.

AN INTEGRATION APPROACH OF CMM WITH CAD/CAM SYSTEMS

R. R. Naqasha, G. A. Al-Kindi, M. J. Kadhim
Dept. of Mech. Eng., The Univ. of Technology, Baghdad, Iraq

ABSTRACT

This paper investigates the level of integration which can be employed between CMM and CAD/CAM systems. Data acquisition was achieved using a particular CMM and a PC. Performance assessment tests were conducted to evaluate the potential of the equipment used. A set of data processing algorithms were developed including; data filtering, automatic resampling, data smoothing and odd points elimination to enable integration with CAD/CAM systems. Further processing algorithms were also proposed and implemented to establish CAD/CAM interface. Conclusions drawn assure that wide industrial applications are possible within the present capabilities.

KEYWORDS

Coordinate Measuring Machine (CMM), Computer Aided Design, Computer Aided Manufactur

1. INTRODUCTION

The integration of manufacturing elements employing computers and machine intelligence philosophy is perhaps the leading way to achieve an "automated factory". For an automated manufacturing cycle, it is vital to transport data between several machines and computers [1], hence the main objective of this paper is to employ data gathered from a CMM to establish automatic generation of computer aided design (CAD) and computer aided manufacturing (CAM) criteria. Fig. 1 illustrates a flow chart of the work.

An important application of CMM can be realized by integration with CAD/CAM software. This requires consequently a physical interface and controlling software [2], to allow CMM data to be readily used by the CAD system. Connecting the digital measurement capabilities of a CMM with CAD systems presents an approach to computerised reverse engineering to be adequately used for the production cycle.

2. IMPLEMENTATION

A coordinate measuring machine has been physically interfaced to a personal microcomputer using serial interface environment. A sample of measurement output table as captured from CMM is illustrated in fig. 2. The geometry information of a measured part is totally presented in the axes data values (X,Y,Z). Consequently, all other information are automatically removed by application of filtering algorithm.

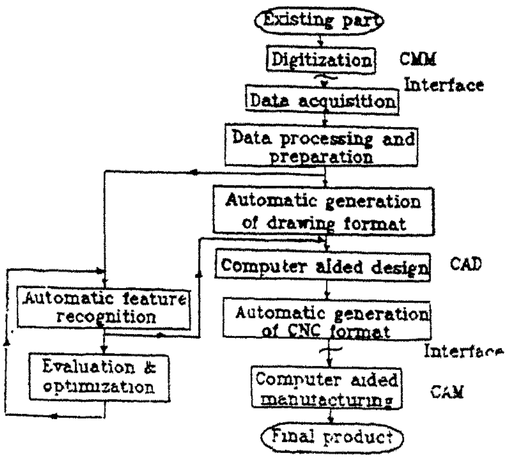


Fig. 1: Global system flowchart

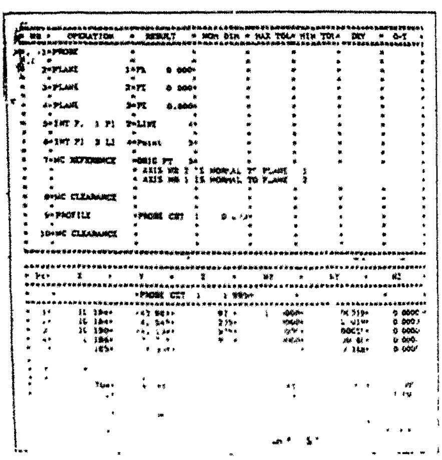
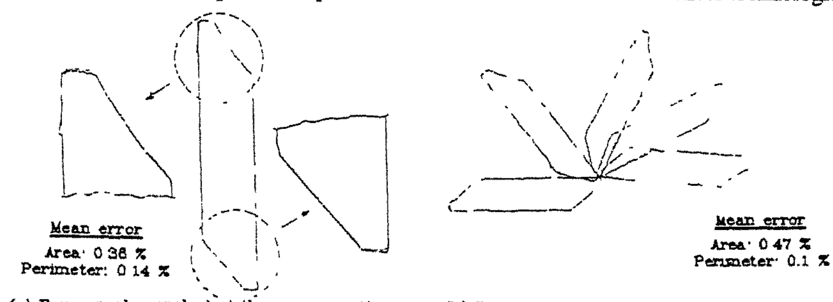


Fig. 2 : A typical example of CMM data.

The CMMs are well known for performing adequately accurate, repeatable and fast measurements [3,4,5]. However, to assess the repeatability of the machine used, two tests were conducted. Firstly, five successive records of a single piece placed in the same position Secondly, repeated records of one piece measured at different rotating angles. For effective comparisons, a uniform, equally numbered set of points was generated by applying a resampling algorithm (see section 3).

Standard deviation (S) has been calculated on two basis. One as the differences in radii, taken from centre of area to each point on the perimeter, and the other is based on differences in resolutions. In both cases, it was found that the values satisfy the range (±2 S) from the mean, which agreed with the machine manufacturer listed specification. Fig. 3 shows a CAD drawing of the samples. Computed feature variation of area and perimeter are given for completness. Resolution changes are within ± 6um. Results assured that CMM's accuracy and repeatability are adequate to be implemented in design (CAD) and manufacture (CAM) by application of reverse engineering. CMMs accuracy, however, may be further improved with continuous and rapid developments in the field of electronic and sensor technologies [6].



(a) Five samples probed at the same position. (b) Five samples probed at different positions

Fig. 3 : CAD drawing of the repeatability test samples.

3. DATA PROCESSING

Data treatment and processing is strongly recommended to handle a number of frequently faced situations during a CMM measuring process, including; overlapped measured points, un-digitised portion of a workpiece feature and non-uniform resolution value.

A set of intelligent algorithms were made to overcome the mentioned cases. First algorithm is devoted to eliminate the overlapped points wherever they exist, it also eliminates blank portions of a digitised contour by inserting mathematically computed points to yield a closed boundary. The importance of the elimination algorithm is to establish a uni-direction closed contour required for further processing, e.g. cutting tool path and area calculation. Fig. 4 shows the flow diagram of this algorithm.

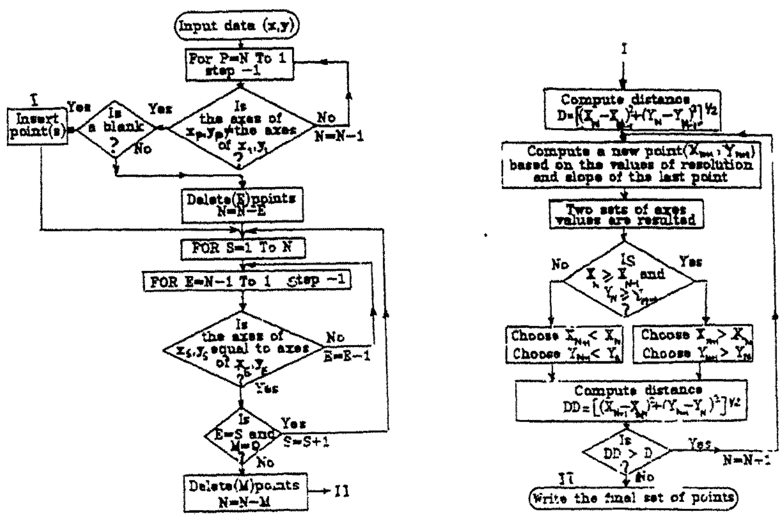


Fig. 4 : Flow diagram of the elimination algorithm

Second algorithm is constructed to create a constant resolution contour to provide homogenuous distribution along the boundary, however, resolution value should be carefully selected to ensure complete definition of geometrical features of the measured part. Fig. 5 presents the logical steps of the developed resampling algorithm.

An additional algorithm was devloped to automatically distinguish and assign outer boundary and inner details. This has high potential application in machining activities, where compensation of tool radius is just the opposite direction for the inner path than for the outer. Flow diagram of the developed boundary recognition algorithm is illustrated in fig. 6. Results of applying the algorithm are presented in fig. 7.

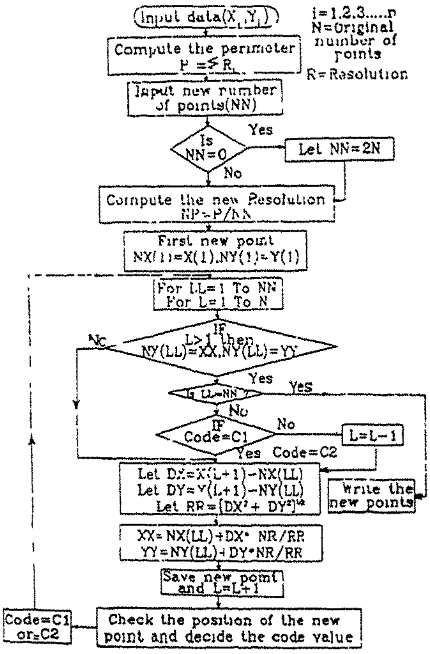


Fig. 5 : Flow diagram of the sampling algorithm.

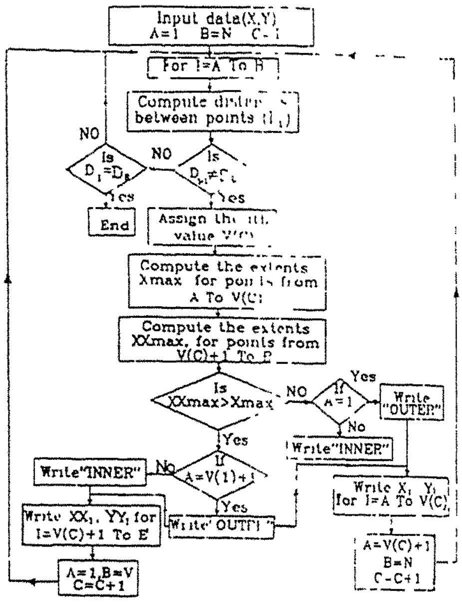
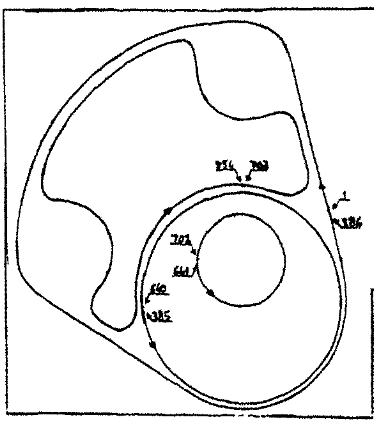
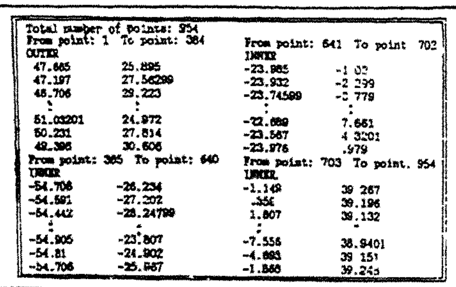


Fig. 6 : Flow diagram of the boundary recognition algorithm.





Total number of points: 954

From point: 1 To point: 384		From point: 641 To point 702	
OUTER		INNER	
47.665	25.895	-23.985	-1 02
47.197	27.56299	-23.932	-2 299
48.706	29.223	-23.74599	-2 779
:	:	:	:
51.03201	24.972	-22.689	7.661
50.231	27.814	-23.567	4 3201
48.396	30.606	-23.976	.979
From point: 365 To point: 640		From point: 703 To point. 954	
INNER		INNER	
-54.708	-26.234	-1.149	39 267
-54.591	-27.202	.356	39.196
-54.442	-28.24799	1.807	39.132
:	:	:	:
-54.905	-23.807	-7.556	38.9401
-54.81	-24.902	-4.893	39 151
-54.706	-25.987	-1.868	39.245

Fig. 7 : A sample workpiece with outer and inner features and its results.

4. DATA SMOOTHING ALGORITHM

Some odd points do exist through an overall set of CMM meaurement due to surrounding environment error, manufacturing capabilities (e.g. surface finish) and inspection limitations. An application to tolerate these odd points was made by applying Fourier Analysis [7,8] The use of this technique has been adopted by the authors to establish an alternative smoothing technique of CMM data. The validity of Fourier series when applied to CMM data could be demonstrated by considering constant digitising speed or uniform sampling of data, thus time intervals are equal. If polar coordinate system is established with workpiece centre of area is set as the reference point of system axes then the set of radii from this centre of area to each point on W.P. boundary could be computed and used confidently in the calculation of Fourier descriptors [9]. To demonstrate such case when applied to CMM data, a typical example is presented using a circular W.P Calculation of uniformly distributed set of points on W.P. boundary using polar coordinate system has been carried out. In addition, a theoretical model of the same size circle was established and same number of points were computed. Fourier descriptors were calculated for the circular W.P. and its theoretical model, fig. 8.

In order to assess the potential of Fourier descriptors in the smoothing of data, reconstruction of object boundary using inverse formula of Fourier transform has been carried out with filtering of high frequencies which may be related to noise and included in the last few Fourier descriptors, however a decision must be taken to point out the exact number of Fourier descriptors which should be tolerated before object reconstruction is made to achieve best smoothing. Object reconstruction of the same circular W.P. was carried out with a different number of tolerated descriptors. Fig. 9 presents a sample of results obtained, where it could be noticed that tolerating of the last 25% descriptors in a set yeilds best smoothing process.

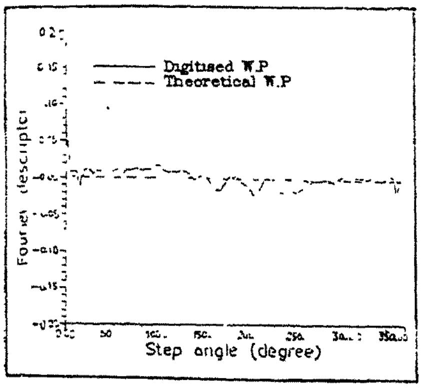


Fig. 8 : Normalized Fourier descriptors of the circular workpiece.

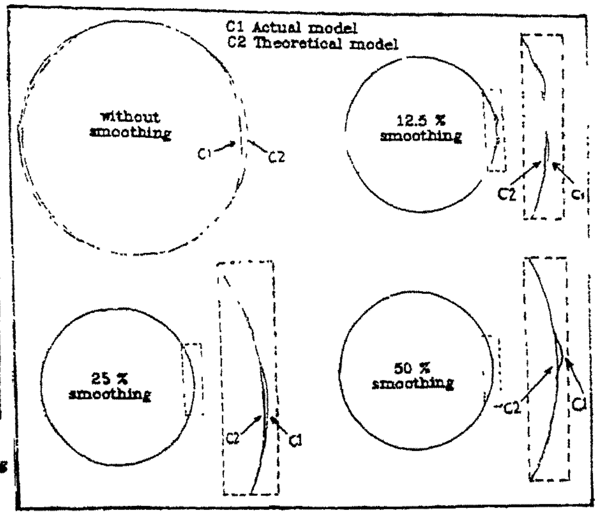


Fig. 9 : An example of different smoothing cases for same reconstructed W.P. (C1).

5. INTERFACING WITH CAD SYSTEMS

Interfacing with CAD systems could be achieved by the generation of a standard data exchange files. DXF file format is frequently used to exchange drawings between CAD systems [10]. In this paper the authors have developed a procedure to automatically create DXF files of workpieces under examination using CMM data gathered. The algorithm manages codes and coordinates values in the relative format and sequence. Two alternative data implementations were used as follow:

a. The drawing is generated as a number of line segments, each consists of two points only sequentialy acquired from a processed CMM data set.

b. The drawing is generated after applying automatic feature recognition procedure[11]. This application was adequately converted CMM data points to basic entity elements namely: Line and arc, which in turn used as input data to the drawing algorithm. Final size of drawing file in this case is much less than the discrete points drawing file.

Figures 10 and 11 present sample results of applying both alternatives.

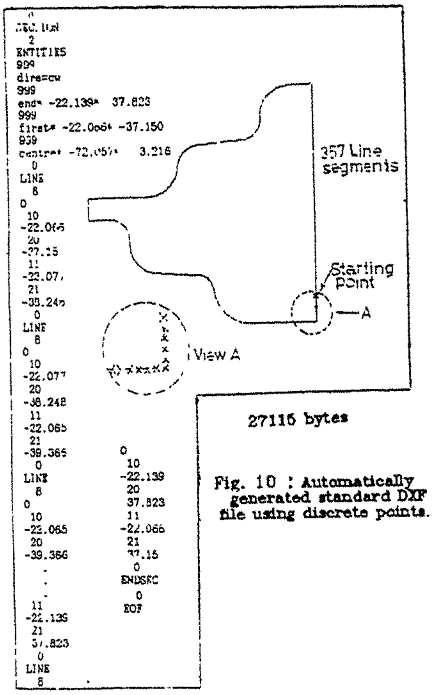


Fig. 10 : Automatically generated standard DXF file using discrete points.

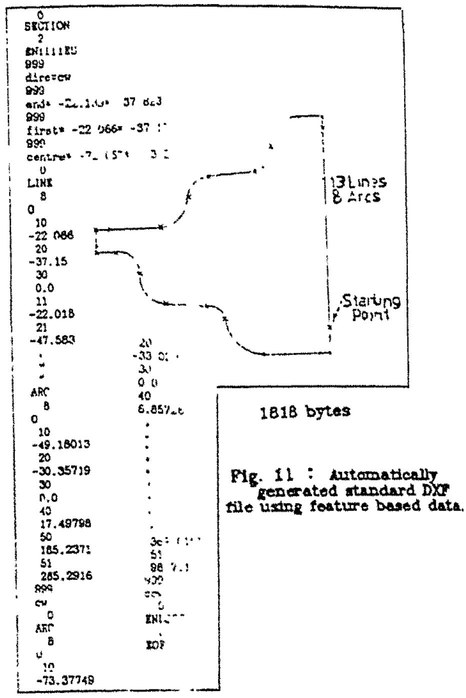


Fig. 11 : Automatically generated standard DXF file using feature based data.

As measured object size may either exceed the viewing area or be too small, the drawing may not appear clearly. The developed algorithm addresses this case automatically by shifting, adjusting and re-scaling of the measured data to find best fitting within the visible area using coordinate values of object boundary points, centre of object area and coordinates of allowed viewing area. Fig. 12 demonstrates example of the automatic positioning and scaling issue.

6. INTERFACING WITH CAM SYSTEMS

An approach to integrate the proposed system with CNC milling machines has been developed. CNC machines are usually operated from suitably coded numerical inputs. CNC program consist mainly of two parts. One part contains the geometrical information that defines the tool path, while the second part involves the cutting parameters namely; program number, units, tool number, feed rate and speed value, coolant, etc.[12]. An algorithm has been designed in this study to generate these codes automatically. The resulted CAM oriented data using the developed algorithm present a complete CNC program and could be readily fed to the milling machine controller.

Two alternatives have been set for data input. First alternative is based on the CMM processed data set. In this case data are presented as small line segments and treated by the algorithm to generate tool path. The output CNC program offers linear interpulations only (G1-code). Each G1 step equals the resolution value as selected in the resampling operation. Fig.13 shows a sample result.

Second alternative is based on processing of CAD DXF file. CAD oriented data can not be used directly for CAM operations, since the drawing format for any workpiece geometry differs from that required for machining program. Certain data processing and sequencing is required to generate cutter location data from the geometry description to be used in the creation of CNC programs. The output CNC program, in this case, offers both linear and circular interpulations (G1, G2, and G3-codes). Fig. 14 presents an example.

7. CONCLUSIONS

In this paper an approach to apply CMM data to CAD/CAM systems has been proposed and investigated. Physical CMM to a PC interface was achieved and CMM data was captured and processed successfully. Data smoothing was investigated using Fourier analysis. Results assured the validity of the technique, however, further investigation for 3D complicated parts is highly recommended to relate the optimum

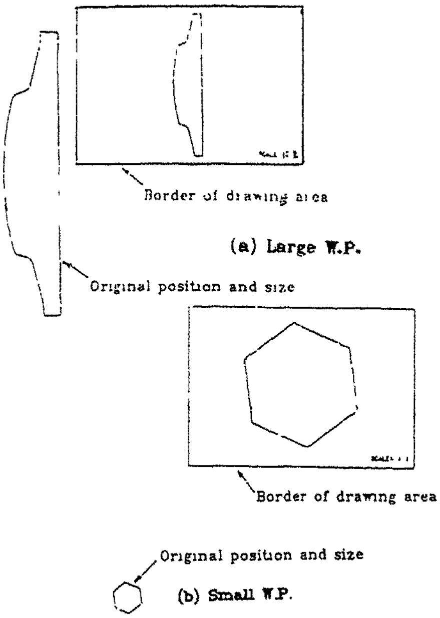


Fig. 12 : Automatic positioning and scaling.

```
%PM
N99319
N1 G71
N2 G18
N3 G52
N4 T1 M67
N5 F150
N6 S 800 M4
N7 M8
N8 G0 X492.93 Z287.858
N9 G0 Y10
N10 G1 Y-3
N11 G43
N12 G1 Z292.858
N13 G41
N14 G1 X482.934
N15 G3 X477.934 Z287.858 R5.00
N16 G1 X 477.982 Z 287.729
N17 G1 X 477.731 Z 288.867
N18 G1 X 477.733 Z 290.004
N19 G1 X 477.735 Z 291.142
N20 G1 X 477.737 Z 292.279
.
.
.
N360 G1 X 477.491 Z 277.171
N361 G1 X 477.499 Z 277.107
N362 G1 X 477.982 Z 277.417
N363 G1 X 477.939 Z 277.894
N364 G1 X 477.933 Z 278.906
N365 G1 X 477.932 Z 280.041
N366 G1 X 477.932 Z 281.152
N367 G1 X 477.929 Z 282.288
N368 G1 X 477.930 Z 283.401
N369 G1 X 477.930 Z 284.514
N370 G1 X 477.935 Z 285.634
N371 G1 X 477.923 Z 286.752
N372 G1 X 477.851 Z 287.858
N373 G3 X 482.861 Z 282.858 R5.00
N374 G1 X 492.861 Z 287.858
N375 G40
N376 G0 Y10 M30
EOT
```

Fig. 13 : Point-to-point CNC program.

```
%PM
N99319
N1 G71
N2 G18
N3 G52
N4 T1 M67
N5 F150
N6 S800 M4
N7 M8
N8 G0 X492.93 Z287.858
N9 G0 Y10
N10 G1 Y-3
N11 G43
N12 G1 Z 292.858
N13 G41
N14 G1 X 482.934
N15 G3 X477.934 Z287.858 R5.00
N16 G1 X 477.[illegible] Y-3.0 Z 277.41[illegible]
N17 G1 X 476.071 Y-3.0 Z 277.167
N18 G1 X 449.314 Y-3.0 Z [illegible]
N19 G2 X 433.397 Y-3.0 Z293.0[illegible] R17.4[illegible]
N20 G3 X 42[illegible] Y-3.0 Z299.7[illegible] R[illegible]
N21 G1 X 40[illegible] Y-3.0 Z [illegible]
N22 G2 X 396.3[illegible] Y-3.0 Z313.791 R[illegible]
N23 G3 X 386.5[illegible] Y-3.0 Z323.[illegible] R[illegible]
N24 G1 X 378.[illegible] Y-3.0 Z [illegible]
N25 G1 X 377.[illegible] Y-3.0 Z 323
N26 G1 X 377.[illegible] Y-3.0 Z 332.[illegible]
N27 G1 X 378.1[illegible] Y-3.0 Z [illegible]
N28 G1 X 406.[illegible] Y-3.0 Z 3[illegible]
N29 G2 X 413.7[illegible] [illegible]
N30 G2 X 420.[illegible] Y-3.0 Z[illegible] R10.[illegible]
N31 G1 X 448.[illegible] Y-3.0 Z [illegible].182
N32 G3 X 451.394 Y-3.0 Z3[illegible].809 R [illegible]
N33 G2 X 467.998 Y-3.0 Z379.[illegible]17 R [illegible]
N34 G1 X 477.74[illegible] Y-3.0 Z 379.26[illegible]
N35 G1 X 477.[illegible] Y-3.0 Z [illegible]
N36 G1 X 477 [illegible] Y-3.0 Z [illegible]
N37 G1 X 477.[illegible] Z [illegible]
N38 G3 X 482.861 Z [illegible] R[illegible]
N39 G1 X 492.861 Z 287.858
N40 G40
N41 G0 Y20
```

Fig. 14 : Feature based CNC program.

selected number of tolerated descriptors to parameters associated with manufacturing limitations, e.g. surface finish, to result in accurate separation of signal noise from object geometry. Interfacing with CAD systems were achieved using automatic generation of data exchange format files. The proposed processing technique for workpiece scaling and positioning yeilded accepted results. Adoptation of previously developed automatic feature recognition technique showed advataguous results for integration with CAD/CAM. G-code programs to control CNC machines were produced automatically using either processed CMM data or CAD oriented data. The integration procedures developed in this paper promise satisfactorly results for wide industrial applications within the present capabilities.

REFERENCES

[1] Chadha, B. and Fulton, R. "Design and Implementation of a CIM Information System" Engineering With Computers, Vol. 10, No.1, pp 1-11., 1994.

[2] Kwok, W., and Eagle, P. "Reverse Engineering: Extracting CAD Data from Existing Parts" Mechanical Engineering, Vol. 113, No. 3, pp. 52-55, March 1991.

[3] Groover, M . Fundamentals of Modern Manufacturing: Material, Process, and Systems. Prentice-Hall Inc. 1996.

[4] Inglesby, T. "CMM in CIM" Manufacturing Systems, Vol. 7; No. 10, pp. 18-25, Oct. 1989.

[5] Cowling, G. and Mullineux, G."Toward An Intelligent CAD-CMM Interface" Engineering with Computers. Vol. 5, No. 2, pp. 133-141, 1989.

[6] Al-Kindi, G. and Al-Shamary, T. "Evaluation Study of Cuurent Progress in the Field of Computer Integrated Manufacturing" Paper submitted for publication, J. of Engineering and Technology, U.O.T, Baghdad, Iraq, 1996.

[7] Al-Kindi, G., Baul, R. and Gill, K. "A Comparison of Orthogonal Transforms in Engineering Computer Vision" Proc. Instn. Mech. Engrs. Vol. 204, Part C, pp. 321-327, 1990.

[8] Al-Kindi, G , Baul, R., and Gill, K. "An Example of Automatic 2D Component Inspection Using Computer Vision" Proc. Instn. Mech. Engrs. Vol. 205, Part B, pp. 71-83, 1991.

[9] Sydenham, P. and Hanckock, N. Introduction to Measurement Science and Engineering, Wiley & Sons Ltd. 1989.

[10] Jackson, S., and Mittal, R., "Automatic Generation of 2-axis Laser-Cutter NC Machine Program and Path Planning from CAD", Computers in Industry, Vol 21, No. 2, pp. 223-231, Feb. 1993

[11] Naqasha, R., Al-Kindi, G and Kadhim, M " Automatic Entity Recognition of Object Features for CMM-CAD/CAMIntegration" Paper submitted and under review to the 32nd Int. Matador Conference to be held on 10-11 July 1997, UMIST, England

[12] Stanton, G., Numerical Control Programming Willy & Sons Inc 1988

A NEW FAST ALGORITHM FOR ENGINEERING DRAWING VECTERIZATION

C.Y. Liu*, Y.H. Chen*, X.M.Liu**, J.R.Liu** and H.T.Li**

*Department of Mechanical Engineering, The University of Hong Kong, Hong Kong

** Software Center, Northeastern University, Shenyang 110006, P.R. China

ABSTRACT

This paper presents a generalized layer Hough transform (GLHT) and a fast augmented linear regression (FALG) for polyline detection in engineering drawing vectorization. In the proposed method, a scanned digital image is first preprocessed and thinned. Then GLHT is used to detect the most important line directions in the image. These directions are followed to extract segments. In the subsequent tracing procedure, FALG is used to correct the tracing direction without consuming too much CPU time Whenever a segment longer than a threshold is found, points within 1 pixel along the segment are deleted. For the remaining image, the above procedure is repeated until no segment longer than 2 (Pixels) is detected. Finally, segments are linked to polyline according to some mathematical methods and pre-defined thresholds.

KEYWORDS

Engineering drawing, image processing, line detection, generalized Hough transform, linear regression.

1. INTRODUCTION

Extracting polyline in line-drawings has been a problem which is not solved thoroughly [1,2], partly because of noises in the source drawing and the scanned digital image. Even though there are some algorithms for line extraction, many of them are noise sensitive. Hough transform (HT) has been doing well as a tool to detect object with or without analytic expression, but there is an unavoidable problem where the higher the precision, the longer the computing time.

There are many improved methods to overcome the problems in both the HT and the Generalized Hough transform (GHT). S.C. Jeng [3] proposed a fast generalized Hough transform (FGHT) for detecting some small elements in a not very large image. At the same time, some hierarchical HTs use subpatterns or gradient information of edge points instead of the original edge points [2,3]. Because the objective is to obtain accurate gradient information, especially for a large and noisy image, such transforms do not work well. So, a new method, called generalized layer Hough transform (GLHT) to detect rough main directions, is proposed in this paper. A new method called fast augmented linear regression (FALG) is used to correct the rough main direction when tracing segment along the rough main directions to extract segments. For any sufficiently long segment (no less than 6 pixels), the precise direction of the segment can be found and the correct segment can be extracted from the scanned image. Finally, by using a fast searching algorithm for segment linking, segments with distance shorter than an experimental threshold, such as 2 pixels, are linked into polylines. A vectorized drawing can then be generated.

In order to make the problem simpler and more efficient while without loss of generality, the scanned grey-level image is converted to a binary image through thresholding. The binary image is then processed by thinning in a way similar to [4]. The proposed generalized layer Hough transform and line segment tracing can then work on such preprocessed images.

2. GENERALIZED LAYER HOUGH TRANSFORM

As is generally known, GHT has been a useful tool for image processing and understanding [2,5,6], but it is a time and storage consuming algorithm. Even though some improvements have been made previously [5-8], the problem is still not solved.

After careful analysis on line drawings, for example, engineering drawings, and district drawings, we find that in most cases, most of the line segments are along only some definite directions. We call these definite directions main directions of the image.

In the GLHT, the main directions are found step by step, that is, the θ space is first divided into some rough intervals, such as 0^0, 20^0, 40^0, 60^0, 80^0, 100^0, 120^0, 140^0, 160^0. The GHT is performed on these intervals. We call this the first layer of GLHT. And then a main direction θ^* of this layer is found, the neighborhood of θ^* is divided into smaller parts, the GHT is performed on the new intervals in the neighborhood of θ^* again. We call this the second layer of GLHT. This procedure is repeated until there is no need to do so.

In most cases, the image is very large, such as the scanned A0 or A1 size of district line-drawing of a city, it may be as large as 10240*10240 pixels. So the image is divided into small blocks, with each block the size of about 1024*1024 pixels.

Let $f(x,y)$ be the grey value of the pixel at the xth row and the yth column. After preprocessing, $f(x,y)$ can only be 0 or 1, with $f(x,y)=1$ representing a black point and $f(x,y)=0$ representing a background point. Let $H(\rho,\theta)$ be the accumulation array of Hough transform space, then the array $H(\rho,\theta)$ is calculated by the following procedure:

```
for (x=0; x<X; x++)
   for (y=0; y<Y; y++)
{
    if (f(x,y)== 1)
   {
          for (θ=0⁰; θ<180⁰; θ+=10⁰)
          {
                 ρ=xcos(θ)+ysin(θ) ;
                 H(ρ,θ) ++ ;
          }
   }
}
```

where, X is the block width and Y is the block height; θ is an angle within $[0^0, 180^0)$. In the proposed algorithm, there are only 18 values of θ with every 10^0 increment. ρ is the distance from the origin point of the 2D Euclidean space to the line with angle θ.

Then, the maximum value of $H(\rho,\theta)$ is searched with different ρ and θ in the (ρ,θ) space. Let $H(\rho_0,\theta_0)=\text{Maximum}\{H(\rho,\theta) \mid 0<\rho<\sqrt{X^2+Y^2},\ 0^0\le\theta<180^0\}$, where angle θ_0 is called the main direction of the first layer. Then for $\theta_0-5^0\le\theta\le\theta_0+5^0$, this interval is divided into 10 parts. That is, there is only 1 degree increment between every two adjacent directions. Now, let $H(\rho^*,\theta^*)=\text{Maximum}\{H(\rho,\theta) \mid 0<\rho<\sqrt{X^2+Y^2},\ \theta_0-5^0\le\theta\le\theta_0+5^0\}$. Angle θ^* is called the main direction of this layer (the second layer). In our case, angle θ^* is called the main direction of the current circle step. In other cases, if it is necessary, the GHT can be done on more layer. Then line segment detection along this main direction is performed. Meanwhile, the FALR algorithm described in the next section is used to correct the tracing direction. Pixels are deleted on or close to the detected segments. It is obvious that when all the pixels along the extracted line segments are deleted, the remaining pixels in the image must be much less than the original one [9].

For the remaining image, repeat the above procedure until no more line segments longer than 2 pixels can be detected.

3. SEGMENT TRACING AND FAST AUGMENTED LINEAR REGRESSION

When the main direction of the image has been detected by GLHT, the next step is to extract line segments along the main direction. But as we all know that the main direction can only be somewhat roughly one. As in the 2D space, the angle θ within $[0^0, 180^0)$ can only be divided into finite numbers for the sake of saving computer memory, such as 18 in our system. Whatever the depth of division (layers as defined in this paper) is, main directions can only be approximate. That is, even though one can take more steps (layers in our words) to get more exact main direction, it is still a approximate direction. So, to get an exact direction of the object to be detected, we adopt the following algorithm.

First of all, for the θ^* got by the above GLHT, we have

if $45^0 \le \theta^* \le 135^0$ then $y=a+bx$; else $x=a+by$;

here,

$$a=\rho/\sin(\theta^*), \qquad b=-1/\tan(\theta^*) \tag{3.1}$$

In normal case, that is when using the standard linear regression, the parameters of the above regression equation are:

$$b = \frac{\sum_{i=1}^{n}(x_i y_i - \bar{x}\,\bar{y})}{\sum_{i=1}^{n}(x_i^2 - \bar{x}^2)} \tag{3.2}$$

$$a = \bar{y} - b\bar{x} \tag{3.3}$$

Here,

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i, \qquad \bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i \tag{3.4}$$

It is obvious that the above calculations are time consuming in the tracing process. So the new algorithm called fast augmented linear regression is proposed to calculate the parameters a and b in the regression model.

Let a_k and b_k be the estimated value of a and b in the kth step respectively, and a_0 and b_0 be the calculated value in the first correct step using (3.1). Let

$$S_k = \sum_{i=1}^{k} x_i^2 \qquad T_k = \sum_{i=1}^{k} x_i y_i \qquad X_k = \sum_{i=1}^{k} x_i \qquad Y_k = \sum_{i=1}^{k} y_i \tag{3.5}$$

then for the $(k+1)$th step we have

$$S_{k+1} = S_k + x_{k+1}^2, \quad T_{k+1} = T_k + x_{+1} y_{k+1}, \quad X_{k+1} = X_k + x_{k+1}, \quad Y_{k+1} = Y_k + y_{k+1} \tag{3.6}$$

so,

$$b_{k+1} = \frac{(k+1)S_{k+1} - X_{k+1}Y_{k+1}}{(k+1)S_{k+1} - X_{k+1}^2} \tag{3.7}$$

$$a_{k+1} = \frac{1}{k+1}(Y_{k+1} - b_{k+1}X_{k+1}) \tag{3.8}$$

Here, according to experience, we let $k>6$, that is we have $a_6 = a_0$, $b_6 = b_0$ only when there are more than 6 pixels on or close enough to the line segment along the original main direction.

From equations (3.5) and (3.6), it is very easy to find that S_k, T_k, X_k, Y_k are the accumulators of x_i^2, $x_i y_i$, x_i, y_i respectively. They are very easy to be calculated. This is what the word augmented means.

Along the direction dictated by the new linear equation, the line segment can be traced until there is no pixel on or close enough to the segment.

4. POLYLINES LINKING THE DETECTED SEGMENTS

Finding polyline which link all the segments has been a time consuming problem, as there are too many segments in a large image, such as, engineering drawing size A0. One has to traverse every segment to find linkings.

In the system, as direction information is already attached to the segments, we need not to consider the segments with almost the same direction. That is, along the same main direction. The algorithm is as the following:

```
for (i=0 ; i < m-1; i++)
    for (j=i+1; j<m; j++)
        for (s=0; s<n_i; s++)
            for (t=0; t<n_j; t++)
{
  if       (distance(segment_start[i][s], segment_start[j][t]) < distance_threhold)
               link( );
  else if  (distance(segment_start[i][s], segment_end[j][t])  < distance_threhold)
               link( );
  else if  (distance(segment_end[i][s],  segment_start[j][t]) < distance_threhold)
               link( );
  else if  (distance(segment_end[i][s],  segment_end[j][t])  < distance_threhold)
           link( );
}
```

where, m is the number of main directions and n is the number of segments along the current main direction; "segment_start[i][s]" and "segment_end[i][s]" are the starting and ending points of the sth segment along the ith main direction respectively; "distance_threshold" is a distance threshold, and it is set to 2 in our experiments. So if the distance between any ends of the segments along different main directions is shorter than distance_threshold, they can be linked into polyline. At the same time, the current linked segment is deleted from the segment group. The final polylines are all the polylines linked by the above procedure and all the segments which have no connection with others.

According to the combination mathematics, and if n_i=n the calculation times for polyline linking is $C_m^2 C_{2n}^1 C_{2n}^1 = 2m(m-1)n^2$ when using our method. When using the ordinary traverse method the calculation time is $C_{mn}^2 C_2^1 C_2^1 = 2mn(mn-1)$. So, by using our polyline searching method, $2mn(mn-1) - 2m(m-1)n^2 = 2mn(n-1)$ calculations are saved. For example, if there are 10 main directions in the image, and about 100 segments per direction, $2mn(mn-1)$=198,000 calculations are saved.

As there is direction message in every detected segment, we only need to consider the segment with different directions. In actual applications, many segments are often found along a few main directions only. With the proposed method, a lot of computation can be saved.

Fig. 1 shows a scanned image of a district map. After preprocessing, segment detection and polyline linking, the image is vecterized. Fig. 2 shows the final output.

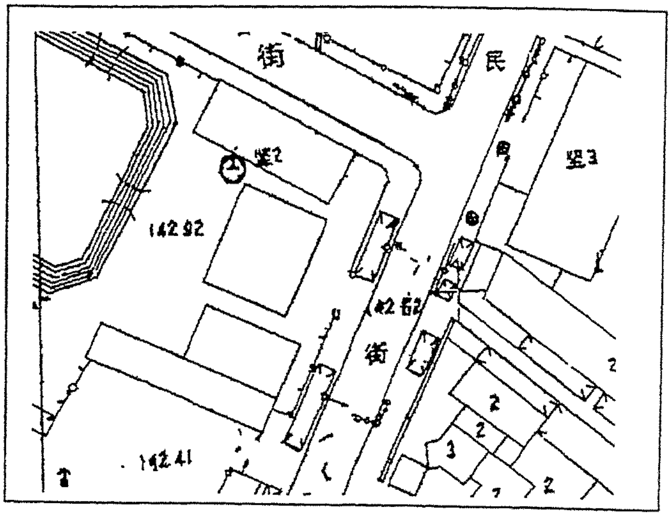


Fig. 1 : One block of the scanned source image

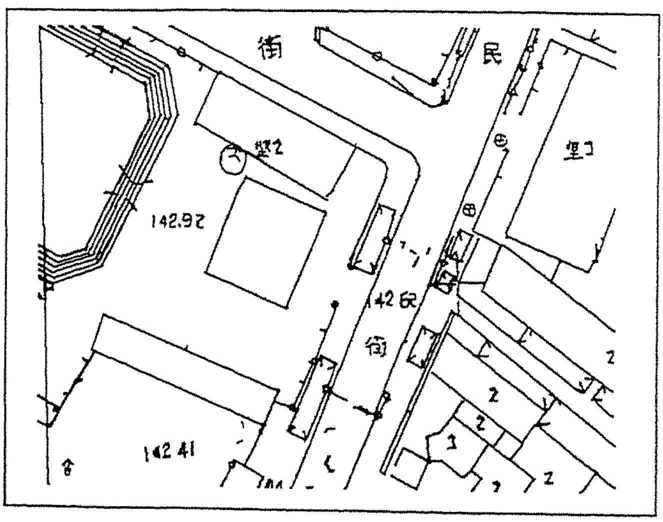


Fig. 2 : The final vecterized drawing of the above block

5. SUMMARY

Based on the GLHT and FALR, we implemented a polyline extraction system for noisy drawings. Our system uses the half-tone scanned digital image as input. Since the GLHT and FALR are noise insensitive, our system can perform very well even for dirty drawings, such as old blue prints of city district and simple engineering drawings.

One of the advantages of our method is that it needs less memory space and less computing time. Another advantage is that, it is not difficult for this method to be used to any element with a high order polynomial mathematical model, such as quadratic or higher curves.

REFERENCES

1. Richard, O.D., Duda, O. and Peter, E.H., "Use of the Hough transformation to detect lines and curves in pictures", Commu. of the ACM, Vol 15, pp.11-15, 1972.
2. Jeng, S.C. and Tsai, W.H., "Fast generalized Hough transform, Pattern Recognition Letters", Vol. 11, pp. 725-733, 1990.
3. Jeng, S.C. and Tsai, W.H., "Scale and orientation-invariant generalized Hough transform - a new approach, Pattern Recognition", Vol.24, No. 11, pp.1037-1051, 1991.
4. Kumar, P., Bhatnagar, D. and Rao, P.S., "Pseudo one pass thinning algorithm", Pattern Recognition Letters, Vol. 12, pp. 543-555, 1991.
5. Conker, R.S., "A dual plane variation of the Hough transform for detecting non-concentric circles of different radii", Comp. Vision, Graphics, and Image Processing, Vol. 43, pp.115-132, 1988.
6. Nagasamy, V. and Langrana, N.A., "Engineering drawing processing and vectorization system", Computer Vision, Graphics, and Image Processing, Vol. 49, pp. 379-397, 1990.
7. Joseph, S.H., "Processing of engineering line drawings for automatic input to CAD", Pattern Recognition, Vol.22, No.1, pp 11-17, 1989.
8. Rosin, P.L., "Techniques for segmenting image curve into meaningful descriptions", Pattern Recognition, Vol. 24, No. 7, pp.643-652, 1991.
9. Liu, C.Y., J.R. Liu and Li, H.T., "Analysis of engineering drawing and generation of its interface with AUTO CAD", Computer Aided Engineering (In Chinese), Vol. 3, No.3, pp.59-63, 1994.

A GOUGING-FREE AND COLLISION-FREE TOOLPATH GENERATION ALGORITHM FOR A ROBOTIC CUTTING SYSTEM

W.C. Tse, Y.N. Hu and Y.H. Chen
Department of Mechanical Engineering
The University of Hong Kong
Hong Kong

ABSTRACT

In this paper, a method that can generate gouging-free and collision-free toolpath for a robotic cutting system is presented. A solid which is enclosed by parametric surfaces can be handled. The method consists of two parts. The first part is a Cutter Location (CL) point generation algorithm which generates optimal Cutter Contact (CC) points according to two criteria namely chordal deviation and scallop height, then the corresponding CL points are generated and relocated to avoid possible gouging. The second part is a Cutter Orientation (CO) selection algorithm that find a collision-free CO for each CL point generated.

KEYWORDS

Toolpath Generation, Robot, Machining, Freeform Surface

1. INTRODUCTION

In recent years, applying robotic cutting system to machine complex surfaces has become a new research area [1][2]. Since robotic cutting system is more flexible than CNC machines in term of tool access, it can handle more complicated geometry. In addition, with the same size of workspace, the robotic cutting system has a 40% of floor space reduction over ordinary CNC devices as stated in [3]. However, such flexibility also adds more difficulties for generation of optimal toolpath. Therefore in this paper, a CL and CO generation method for robotic cutting system is presented.

The proposed system consists of a six axis articulated robot holding a spindle which in turn rotates a milling cutter. When applying this system to machine a parametric surface of a freeform solid, conventional CNC methods for toolpath generation are not adequate since additional factors need to be considered, especially the possible occurrence of collision when the robot arm trying to move the cutter towards the desired CL point. Our toolpath algorithm is divided into two parts: (1) The generation of gouging-free CL point would be discussed in section 2;(2) The selection of collision-free CO is described with in section 3.

2. CUTTER LOCATION POINT CALCULATION

Our CL point generation algorithm which is a remedy for the algorithm of Faux and Pratt[4] is as follows: The toolpath is generated by moving the cutter along the u direction of the parametric surface, and the distance between two passes is determined from the parameter v. Optimal number of CC points are generated from the parametric surface according to the machining tolerance. The number of CC points used is a critical factor to the machining process. Rough surface would be resulted if too few CC points are used, on the other hand, too many CC points would consume large storage space, long computation and machining time.

Two indices are used to control the density of the CC points to an optimal level:

- Chordal Deviation (CD) controls the forward-step size in the isoparametric curve direction (u)
- Scallop Height (SH) controls the side-step size in the isoparametric curve direction (v)

The corresponding CL point for each CC point is determined. These CL points are then relocated to avoid gouging caused by connecting them by line segment.

2.1 Forward-Step Determination

A straight line segment is drawn from *p(0)* to *p(1)* and the Chordal Deviation (CD) can be calculated (Figure 1). If the calculated CD is greater than the machining tolerance, a new end point $p(r_i(u))$ is used instead of *p(1)* and the curve is subdivided into two segments. The subdivision process is continued until the CD for all straight line segments are within the machining tolerance.

When calculating the CD, first $|\vec{d}(u)|$ which is the distance of an arbitrary point on the curve *r(u)* to the straight line $\bar{p}(1) = p(1) - p(0)$ must be computed. Since the patch definition for the freeform surface exists, by holding *v* constant, a parametric curve equation $r_i(u) = r(u, v_i)$ can be obtained.

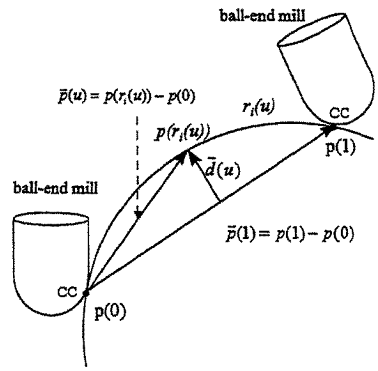


Figure 1. Chordal deviation

Before calculating the maximum $|\vec{d}|$, the condition, u=0 and u=1 at the respective end points must be given first. Any curve segment between $u_0 \le u \le u_1$ can be transformed to a new parameter u', i.e.

$$\begin{aligned} u &= (1-u')u_0 + u'u_1 \\ &= u_0 + (u_1 - u_0)u' \\ &= u_0 + \Delta u_0 u' \end{aligned}$$

where $\Delta u_0 = (u_1 - u_0)$. Moreover, if the parameter u has polynomial form, then it can be written as follows:

$$u^n = \sum_{r=0}^{n} \frac{n!}{(n-r)!\,r!} u_0^r (u'\Delta u_0)^{n-r}$$

If the chord between *p(0)* and *p(1)* is represented by vector $\bar{p}(1)$ (Figure 1), let $\vec{d}(u)$ represent the perpendicular vector of $\bar{p}(1)$ to an arbitrary point $p(r_i(u))$ of the curve, then its equation $\vec{d}(u) = \bar{p}(u) - \lambda\bar{p}(1)$ can be obtained, where λ is the proportional coefficient of the vector $\bar{p}(1)$.

Since $\vec{d}(u)$ and $\bar{p}(1)$ are orthogonal, i.e. $\vec{d}(u) \cdot \bar{p}(1) = 0$, using this equation to solve for λ, then yields:

$$\vec{d}(u) = \bar{p}(u) - \frac{\bar{p}(u)\cdot\bar{p}(1)}{|\bar{p}|^2}\cdot\bar{p}(1) \qquad (1)$$

Moreover, if vector $\bar{p}(1)$ is written in the form of an array, according to the nature of vector, $\vec{d}(u)$ can be further simplified as:

$$\vec{d}(u) = F\bar{p}(u) = (1 - \bar{p}^T(1)\cdot\bar{p}(1) / (\bar{p}^T(1)\cdot\bar{p}(1)))\cdot\bar{p}(u) \qquad (2)$$

where $F = 1 - \bar{p}^T(1)\cdot\bar{p}(1) / (\bar{p}^T(1)\cdot\bar{p}(1))$ is called the projection matrix.

To calculate the maximum perpendicular vector $|\vec{d}(u)|_{\max}$, the curve's tangent $\dot{r}_i(u)$ needs to be calculated first and its vector form can be written as $\vec{T}_r = (\dot{r}_i(u),\ 1)$. Then the u_{max} that brings about $\vec{d}(u)_{\max}$ can be obtained by solving the following equation:

$$\vec{d}_{\max} \cdot \vec{T}_r = F\vec{p}(u_{\max}) \cdot \vec{T}_r = 0 \qquad (3)$$

The polynomial equation (3) is of order *2n-1*, but for a curve of degree *n*, actually only *2n-3* roots must be solved

In summary, the procedure to determine CC data for machining a parametric surface can be briefly described as follows: Starting at one corner of the surface, a patch is loaded, v is held at one value while chords are approximated in the u direction. Then the CC points located during the subdivision process are stored temporarily and will later be used in the representation of the isoparametric curve by straight line segments. When the last chord approximating $r_i(u, v_i)$ is detected, a side step in the v direction or a new patch is loaded. This procedure will be continuously carried out, until all patches are exhausted.

2.2 Side-Step Determination

Scallop Height (SH) is dependent on tool shape. For the case of a ball end mill, it can be calculated based on cutter radius and side-step size (v direction step size). For a line segment, the SH can be calculated with equation $R^2 - (R - h_i)^2 = (l_i/2)^2$, where R is the ball end mill radius, h_i is the scallop height, and l_i is the side-step size. For a curve segment $r_i(v)$, the equation $\Delta = \sum_{i=1}^{n} \Delta_i = \sum_{i=1}^{n} l_i / S_i(v)$ can be used to determine the scallop height, where $S(v) = \int_0^v \sqrt{x^2(v) + y^2(v) + z^2(v)}dv$, can be used to judge the approximating degree of the original curve by the cutting path.

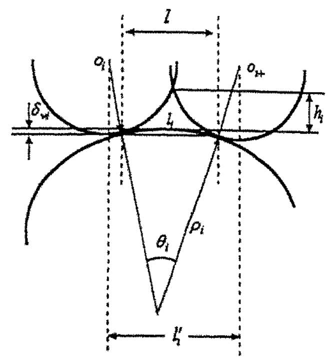


Figure 2. Scallop Height Calculation

Moreover, for small steps, the curve may be approximated by its osculating circle with radius ρ_i. The chordal deviation for a side step based on SH can be calculated using following equation.

$$\delta_{vi} = \rho_i - (\rho_i^2 - (l_i'/2)^2)^{1/2} \qquad (4)$$

where l_i' is the chord length (Figure 2). Since the arc length $\widehat{l}_i$ is related to the angle between the end points of chord l_i'

$$\widehat{l}_i = \rho_i \theta_i \qquad (5)$$

and the chord length is

$$l_i' = 2\rho_i \sin(\theta_i/2) = 2\rho_i \sin(\widehat{l}_i/(2\rho_i)) \qquad (6)$$

then chordal deviation is found by substituting equation (6) into equation (4)

$$\delta_{vi} = \rho_i[1 - \cos(\widehat{l}_i / 2\rho_i)] \qquad (7)$$

where $\widehat{l}_i/(2\rho_i) \approx l_i/(2\rho_i) << 1$. If writing $\cos(\widehat{l}_i/(2\rho_i))$ in Taylor's series, it may be approximately expressed by:

$$\cos(l_i/(2\rho_i)) \approx 1 - l_i^2/(8\rho_i^2) \qquad (8)$$

So combining equation (7) and (8), the equation below yields:

$$\delta_{vi} = \frac{h_i(2R-h_i)}{2\rho_i} \qquad (9)$$

Normally, $l_i' < 2R$ or h_i<R could make the surface be machined completely. Moreover, for R<ρ_i, the gouging between the cutter and the surface can be avoided, therefore before selecting the cutter for machining a specified surface, the curvature of the surface should be checked first. Only if the side step is small enough, then the surface can be milled precisely. In addition, δ_v<h_i also should be satisfied so that the chordal deviation on a side step is less than the given scallop height.

The arc length may take different u at different positions of curve $r(u_i, v)$, so the greatest number of passes is calculated to achieve the given machining tolerance. In practice, if the patches are severely deformed, more checks for arc length is worth reducing the machining time.

2.3 CL Point Relocation

After obtaining the CC points, the next step is to generate the CL points. For the given curve $r_i(u)$, assume that two adjacent CC points (cc_1 and cc_2) and the corresponding CL points (o_1 and o_2) are known and the machining tolerance is given by d, then when the cutter centre moves along a line from point o_1 to point o_2, the tool tip will move from a to b and the surface may be overcut. The shadow region between the lines cc_1cc_2 and ab (Figure 3) is the gouging or interference region. Therefore, it is necessary to relocate the cutter centre o_1 and o_2 so as to avoid such gouging.

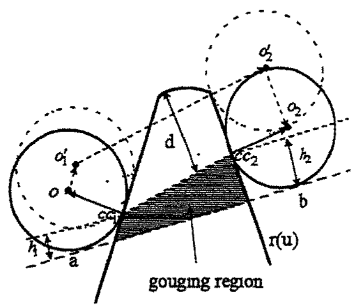


Figure 3. Gouging Elimination

The following algorithm is used to generate the new CL points. In this algorithm (Figure 3) let o_i and $\vec{n}_i$ (i=1,2) be the original CL points and unit normal vectors respectively. Then the amount of gouging h_i at cc_i are given as

$$h_1 = R(1-(1-(\vec{n}_1 \cdot (-\vec{o}_d)^2)^{1/2}) \qquad (10)$$

$$h_2 = R(1-(1-(\vec{n}_2 \cdot (\vec{o}_d)^2)^{1/2}) \qquad (11)$$

To eliminate this gouging, the point o_1 and point o_2 must be relocated along the tangent direction of $r_i(u)$ at the corresponding CC points. Suppose the relocated points are o_1' and o_2', and let $\vec{o}_d = (o_2 - o_1)/|o_2 - o_1|$, $\vec{o}_d' = (o_2' - o_1')/|o_2' - o_1'|$, and $\vec{s}_d = (p_2 - p_1)/|p_2 - p_1|$, then the following equations can be used for calculating the new CL data.

$$o_1' = o_1 + |o_1' - o_1| \frac{(\vec{n}_1 + \vec{s}_d / (\vec{n}_1 \cdot (-\vec{s}_d)))}{|\vec{n}_1 + \vec{s}_d / (\vec{n}_1 \cdot (-\vec{s}_d))|} \qquad (12)$$

$$o_2' = o_2 + |o_2' - o_2| \frac{(\vec{n}_2 - \vec{s}_d / (\vec{n}_2 \cdot \vec{s}_d))}{|\vec{n}_2 - \vec{s}_d / (\vec{n}_2 \cdot \vec{s}_d)|} \qquad (13)$$

Therefore the original CL points o_1 and o_2 should be relocated to o_1' and o_2' in order to avoid the gouging caused by connecting the original CL points.

3. DETERMINE CUTTER ORIENTATION

After the generation of CL point, another necessary task is to determine a suitable CO for each CL point. When a six axis articulated robotic machining system is used to machine a freeform solid, the first factor that affects the CO is the accessibility of the cutter to the given surface. So in order to let a cutter move into a given point, two types of collision should be avoided: (1) Collision between the cutter shank and the stock-in-progress; (2) Collision between the tool-holder and the stock-in-progress;

Besides the collision factor, the second factor that decides the CO is the dynamic characteristics of the machining process for the proposed robotic cutting system. In order to reduce the vibration of the robotic system during machining, the cutter is tilted from the surface normal at the CC point. Practically, both of these factors should be considered when determining the CO. In the following sections, the collision between the cutter shank and the stock-in-progress would be used to illustrate the collision detection and avoidance algorithms. The collision between the tool-holder and the stock-in-progress is omitted, since the same approach can be applied.

3.1 Collision detection

Collision detection involves time consuming volume intersection checking between the stock-in-progress which is modelled by a freeform solid and the cutter shank at each CC point. In order to simplify this process, the freeform solid is approximated by a voxel (cube) representation. Thus the complex interference checking problem between the freeform solid and the cutter shank (a cylinder) is simplified to a much easier interference checking between the voxels (cubes) and the cutter shank (a cylinder).

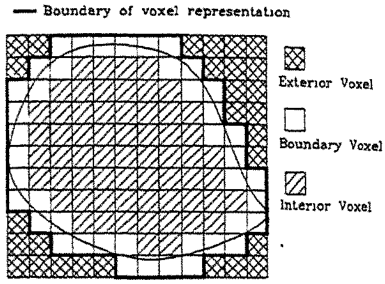


Figure 4.The voxel approximation of a 2D freeform shape.

A voxel approximation of the freeform solid is first generated. There are three kind of voxel, namely interior voxel, exterior voxel and boundary voxel. The interior voxels are completely enclosed by the freeform solid, whereas the exterior voxels are located completely outside the freeform solid. Between the above two, there are boundary voxels which are intersected by the boundary of the freeform solid. The boundary voxels can be obtained by surface sampling. The interior and exterior voxels can be distinguished by the freeform solid definition itself. The freeform solid is approximated by the minimum enclosing voxel set which includes both the interior and boundary voxels (Figure 4). As an illustration, a 2D example of the above approximation scheme is shown in Figure 4.

The checking of intersection between a voxel V and the cutter shank cylinder Q can be further simplified. Instead of finding the volume intersection between V and Q, we check collision by analysing the position of the centre point p of V with respect to the outer offset cylinder O and the inner offset cylinder I of Q [5]. p located outside O means that V is completely outside Q. V is completely inside Q if p resided inside I. Finally if p is inside O but outside I implies that V is intersecting with the boundary of Q. A 2-D illustration of the above process is shown in Figure 5. For 3-D case, the radial offset r and axial offset h used to expand and contract Q to form O and I can be determined from Q', where Q' is a cylinder with the same orientation as Q circumscribing V as shown in Figure 6.

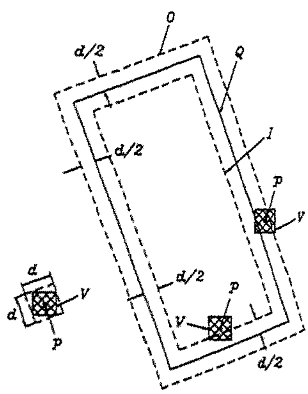


Figure 5.The simplified intersection checking method.

If all the voxels belonging to the voxel representation of the freeform solid (interior and boundary voxels) fall completely outside the cutter shank cylinder Q (as implied by p outside O), there would not be any collision between the freeform solid and Q. On the other hand, collision may occur if there is any voxel belonging to the voxel representation of the freeform solid intersecting with Q (as implied by p inside O above).

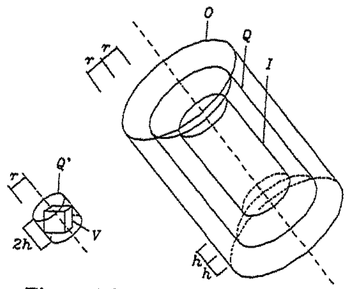


Figure 6. Determination of radial offset r and axial offset h.

3.2 Collision Avoidance

The initial CO $\vec{m}_0$ is set along the surface normal $\vec{n}$ at the CC point. If any collision is caused by $\vec{m}_0$ as detected by the above collision detection algorithm, two stages of cutter rotation are applied to avoid the collision. In the first stage, the cutter is rotated about the cutter tip centre point o along the plane containing $\vec{n}$ and $\dot{r}(u,v_i)$ in both positive and negative direction by an amount of α as shown in Figure 7. The above procedure is repeated with the value of α increased in a stepwise manner (e.g. a step of 5°) from 0° to a limit of 60° until a collision-free CO like $\vec{m}_2$ is identified. If after the first stage of rotation, collision still cannot be avoided, the second stage of rotation needed to be carried out. At this time, the centre of rotation is o and the cutter is rotated about $\vec{n}$ from 0° to 360° in a stepwise manner as shown in Figure 8. The process is repeated for each value of α used in stage one until a collision-free CO is available. If collision still cannot be prevented after the two stages of rotation, the CC point is inaccessible and need to be skipped or modified.

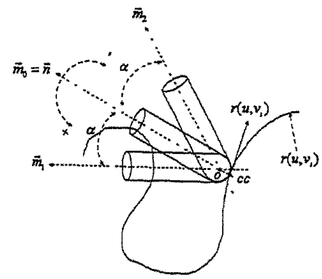


Figure 7.First stage of cutter rotation.

4. CONCLUSION

A gouging-free cutter location point generation algorithm and collision-free cutter orientation selection algorithm have been developed. They can be used together to generate the toolpath for a robotic cutting system which can handle freeform solid enclosed by parametric surfaces.

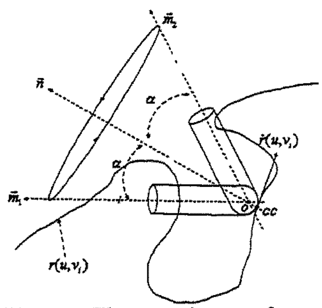


Figure 8. The second stage of cutter rotation.

ACKNOWLEDGEMENT

The research presented in this paper is supported by a RGC grant HKU 567/96E and a CRCG grant 337/064/0040.

REFERENCES

[1] Johan W.H. Tangelder and Joris S.M. Vergeest, "Robust NC path generation for rapid shape prototyping.", Journal of Design and Manufacturing 4 (1994), P281-292.

[2] Johan W.H. Tangelder and Jòris S.M. Vergeest, "Computation of voxel maps containing tool access directions for machining free-form shapes", The 1996 ASME Design Engineering Technical Conference and Computers in Engineering Conference.

[3] Belforte, D.A., "Robotic manipulation for laser processing." Proceedings of the SPIE-High power lasers and their industrial applications. Vol. 650(1986), 262-270.

[4] Faux and M. J. Pratt, Computational Geometry for Design and Manufacture, Ellis Horwood Ltd, Chichester, 1979.

[5] Y.T. Lee and A.A.G. Requicha, "A Family of Algorithms Based on Representation Conversion and Cellular Approximation", Communications of the ACM, Volume 25, Number 9, p642-650, 1982.

ANALYSIS OF ROBOT WORK BY MODELLING THE UNIT MOTION OF ROBOTS

K. S. Kwon and J. S. Kim
Department of Industrial Engineering
Jeonju University
Korea

ABSTRACT

This study discusses the development of robot work measurement method which effectively establishes time standards. It classifies and standardizes the various robot work methods in manufacturing industries. The RObot Modularization of the Unit Motion (ROMUM) is realized by the module of 2 steps unit motions (GET, PUT). This method reduces time and effort of analysis, and can be done with ease.

KEYWORDS

Robot Work Measurement, Modularization, Unit Motion, Predetermined Time Standards

1. INTRODUCTION

The fundamental object of work measurement is to precisely establish the time standards which are the indices of labor productivity. The primary method of work measurement was stop-watch method. It had subjectivity of a human operator and needed to be standardized operator's performance, so the new work measurement method was required to solve these problems. For these requirements, Predetermined Time Standards (PTS) were developed[1,2]. If the patterns of job motions are known, PTS can establish the time standards of job and activity at not only the workplace, but also the laboratory. With these merits, Robot Time and Motion (RTM) was developed in a robot motion and time study, which can predetermine robot cycle times by basing on standard elements of fundamental robot work motions[3,4]. The ROBOT Maynard Operation Sequence Technique (ROBOT MOST) was developed in order to compare the performance of a human with that of a robot in the same task[5]. Also, the ROBOT MODular Arrangement of Predetermined Time Standards (ROBOT MODAPTS) which can analyze a motion through the analysis of the each joint's function of robot was devised[6].

In RTM, ROBOT MOST and ROBOT MODAPTS, manager and operator are restricted by their skills and the time and effort of analysis due to analyzing work into basic motions like Methods Time Measurement (MTM).

Therefore, this study discusses the development of method which is easy to use and effective to set up time standards in order to solve these problems appearing while robot's motion is analyzed. This method can cut down the time and efforts of analysis through the modularization of the unit motions.

2. THE STRUCTURE OF ROMUM

In order to modularize the unit motions, there are five basic motions in ROMUM : reach (R), move (M), grasp (G), pre-grasp/release (PG/PRL), release (RL).

Generally, the robot tasks, whatever robot may handles any objects, keep 2 steps (GET, PUT) which contain basic motions. The ROMUM standardizes the type of the unit motions (GET, PUT) according to the purpose and character of job, and sets the motion modules of each unit motion. The time standards of ROMUM are decided by the modules and moving distance.

Robot tasks can be classified such as pick and place, material transfers, assembly and welding. In the subject of these tasks, the shapes of combination of representative operations are described in Table 1. PG/PRL occurs according to work conditions.

Table 1 : Combination of the basic motions in operation

No	Operation	Combination of the basic motions	
		GET	PUT
1	Reaching hand to start point	R	
2	Arc welding, spray painting while moving	R (PG/PRL)	M
3	Automatic nut supplier task	R (PG/PRL)	RL2
4	Spot welding, pressing the switch	R (PG/PRL) G0	RL0
5	Drawing and pushing with fingers touching	R (PG/PRL) G0	M RL0
6	Picking up parts, and throwing it into chute	R (PG/PRL) G1	M RL0
7	Releasing parts to a different place	R (PG/PRL) G1	M (PG/PRL) RL1
8	Inserting parts	R (PG/PRL) G1	M (PG/PRL) RL2
9	Releasing parts to a different place in complicated condition	R (PG/PRL) G2	M (PG/PRL) RL1
10	Inserting parts to a different place in complicated condition	R (PG/PRL) G2	M (PG/PRL) RL2

The following Fig. 1 illustrates the structure of modularization of the unit motions which are obtained by classifying and standardizing the operations of Table 1.

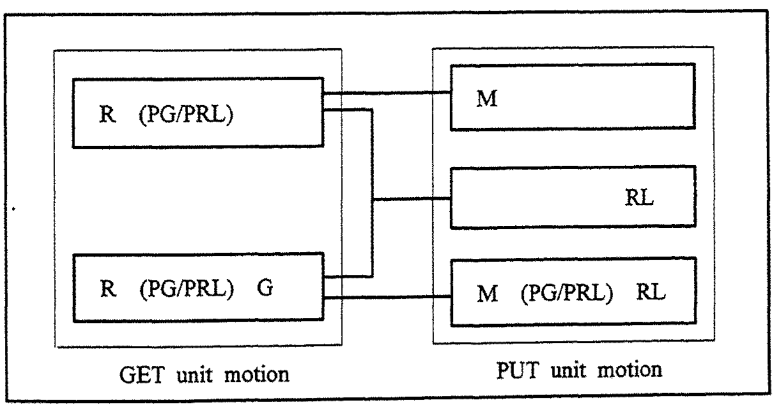


Fig. 1 : Structure of modularization

3. THE BASIC MOTIONS OF ROMUM

3.1 Transport (R/M)

Reach (R) is an action to change the position of end effector (grippers, welding gun, etc.) Move (M) is an action to relocate an object or conduct the operation with predominant purpose while moving.

3.2 Grasp

It is an action which contacts and controls one or many objects. The major variable affecting time values is the type of grasping.

3.2.1 Contact Grasp (G0)

It is an action to completely control an object by contacting with the end effector

3.2.2 Simple Grasp (G1)

It is an action to grasp one object with fingers by a trial. The grasping time depends on the thickness of the object.

3.2.3 Complex Grasp (G2)

It occurs when the object is among the several parts. The end effector moves the object to a different place, and then grasps it. The time values depend on the moving distance and the thickness of the object.

3.3 Pre-Grasp/Release (PG/PRL)

PG/PRL occurs when the end effector needs to change its position before a grasping or releasing action in order to help this action to be performed easily. It consists of pitch, yaw and roll. A simultaneous action can be possible according to the moving distance when it is combined with transport movements.

3.4 Release

It is an action to place the controlled object to the destination. The major variable affecting the time values is the type of releasing.

3.4.1 Contact Release (RL0)

It is an action which is opposed to contact grasp like a throwing or pushing action.

3.4.2 Simple Release (RL1)

It is an action which is opposed to simple grasp, and it is to place the object to a destination by opening fingers.

3.4.3 Insert Release (RL2)

It is an action to insert an object into a hole, and it consists of an inserting and releasing action. Sometimes sliding can occur. The time variable is inserting distance.

(a) Simple Insert Release (RL2A)

It consists of an inserting and releasing action, and usually occurs when it is an inserting task with a simple action.

(b) Complex Insert Release (RL2B)

It occurs when the object is inserted into a target with sliding by the help of an equipment such as a sensor inside or outside, or Remote Center Compliance (RCC). It consists of a sliding, inserting and releasing action.

4. THE MODULES OF THE UNIT MOTIONS

4.1 The Type of GET Unit Motion

As work conditions, four types of the motion modules are established by GET={R, PG/PRL, G}.

4.1.1 GET1 = R

It occurs when the end effector reaches to the start position or work position for arc welding or spray painting.

4.1.2 GET2 = R + G0

It occurs when the end effector contacts with an object or performs spot welding

4.1.3 GET3 = R + G1

It occurs when the end effector grasps one object.

4.1.4 GET4 = R + G2

It occurs when the end effector has difficulty to grasp an object by one trial because the arrangement of the objects is complex.

4.2 The Type of PUT Unit Motion

As work conditions, eight types of the motion modules are established by PUT={M, PG/PRL, RL}.

4.2.1 PUT1 = M

It occurs when the end effector operates arc welding or spray painting while moving.

4.2.2 PUT2 = RL0

It occurs when the end effector contacts with an object for a moment like spot welding.

4.2.3 PUT3 = RL2A

It occurs when the end effector performs a simple insertion task by the help of equipments like an automatic nut supplier through the tube.

4.2.4 PUT4 = RL2B

It occurs when the end effector performs a complex insertion task by the help of equipments like an automatic parts supplier through the tube.

4.2.5 PUT5 = M + RL0

It occurs when the end effector pushes or throws an object.

4.2.6 PUT6 = M + RL1

It occurs when the end effector moves and releases an object like material transfers.

4.2.7 PUT7 = M + RL2A

It occurs when object is inserted into a different place, a comparatively simple insertion task.

4.2.8 PUT8 = M + RL2B

It occurs when the insertion task is impossible by one trial while correcting the location of an object with the help of a sensor inside or outside.

5. THE WORK MEASUREMENT BY ROMUM

As giving each basic motion to conditions, the each time is set up and added by the unit motions to bring out the ROMUM's time data.

The work measurement by ROMUM is executed by the following steps:

(1) divide robot tasks in the field of production into the GET and PUT unit motion.

(2) select the case with moving distance at each unit motion.

(3) select the type of motion modules which is proper to the purpose and character of job from the GET and PUT unit motion.

(4) establish the time standards including conditionally additional motions which occur in compliance with work condition by the unit motions.

6. INSERTION TASK

In order to illustrate the application of the ROMUM method, an insertion task performed by the SCORBOT-ER V robot is described in Fig. 2. It shows the layout of the robot operations.

The robot task in Fig. 2 proceeds 3 steps. Initially, a peg is placed in the fixture and the base part is placed to the side of table. First, the robot picks up the base which is 5 cm distant from the start position and moves it to the work position which is 30 cm distant from first position of the base. Then, the robot picks up the peg which is 20 cm distant from the work position and inserts it into the base. Finally, the robot returns to the start position.

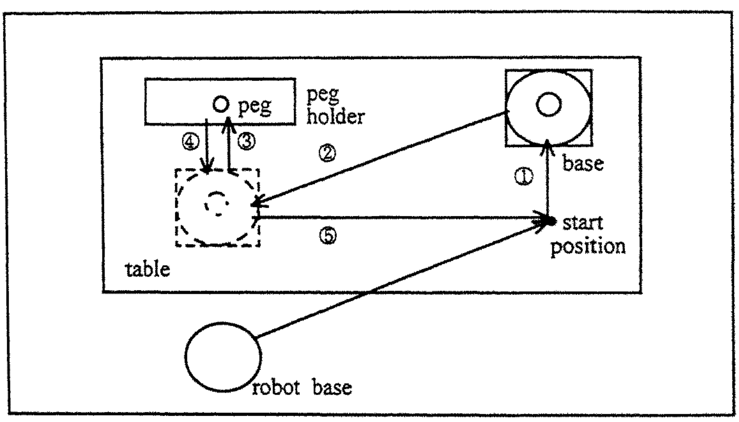


Fig. 2 : Layout of insertion task

Results of the ROMUM analysis are shown in Table 2. The work content of No. 1 is expressed the type and case of GET and PUT unit motion as GET3B and PUT6E respectively. In other words, the GET unit motion consists of moving 5 cm (case B) and grasping an object (R, G1 → type GET3), and the PUT unit motion consists of moving 30 cm (case E) and releasing it to a different place (M, RL1 → type PUT6). Additional motions such as inserting distance, weight or thickness of an object as work conditions in each unit motion did not to be considered. The procedure mentioned above is applied to 2 and 3 steps, and the time standards are established by adding each time. The unit of time used in ROMUM is one hundredth of a second, and is referred to as one Time of Unit Motion (TUM).

Table 2 : The ROMUM analysis results for insertion task

No.	Description	GET unit motion		PUT unit motion		Freq.	Time unit
		Type, Case	Additional motion	Type, Case	Additional motion		
1	Move the base to the work position	GET3B		PUT6E			295
2	Insert a peg into the base	GET3D		PUT7D			322
3	Return to the start position	GET1E					195
Total time							812 TUM 8. 12 sec

7. CONCLUSIONS

As a results of this study, we can modularize the various robot work methods into the unit motions which are standardized, and establish the time standards by selecting the unit motions and the moving distance as variable factors. It will increase the convenience of use for the unskilled worker and decrease the analysis time, cost and errors as comparing with the existing analysis methods.

To realize the ease of use, the minimization of judgement and the reduction of time for analysis, the computer technology as a method of interaction with the user can be applied for developing the robot work measurement in the future research.

8. REFERENCES

1. Barnes, R.M., Motion and Time Study: Design and Measurement of Work, pp.361-389, John Wiley & Sons, New York, 1980.
2. Niebel, B.W., Motion and Time Study, pp.508-586, Irwin, Homewood, Illinois, 1982.
3. Nof, S.Y. and Lechtman, H., "Robot Time and Motion System Provides Means of Evaluating Alternate Robot Work Methods," Industrial Engineering, April, pp.38-48, 1982.
4. Paul, R.L. and Nof, S.Y., "Work Methods Measurement - A Comparison between Robot and Human Task Performance," International Journal of Production Research, Vol.17, No.3, pp.277-303, 1979.
5. Wygant, R.M., "Ergonomics: Robot selection," International Encyclopedia of Robotics, Dorf, R.C.(Ed.), pp.462-477, John Wiley & Sons, New York, 1988.
6. Kwon, K.S. and Lee, S.Y. "A Study on the Robot Motion Analysis by ROBOT MODAPTS Technique", Journal of the Ergonomics Society of Korea, Vol.11, No.2, pp.15-21, 1992.

MOTION PLANNING AND COLLISION AVOIDANCE WITH COMPLEX GEOMETRY

Stephen Cameron and **Caigong Qin**
Oxford University Computing Laboratory
Wolfson Building, Parks Road, Oxford OX1 3QD, U.K.

Stephen.Cameron@comlab.ox.ac.uk
phone: +44 (1865) 273850 fax: 273839

Abstract
CAD and robot simulation tools are often used to verify a workcell or process at the design phase: a geometric model of the machinery involved is produced, and pictures of the model examined for potential problems. However more advanced techniques for collision detection and path planning are only slowly filtering out of the research laboratories. We introduce a path planning shell called OxSim that is based on the separation of *geometry* and *planning*. We claim that OxSim makes it much easier to prototype collision detection and path planning schemes involving large and complex geometries, and outline how that system has been used up to now, and our plans for further research.

Keywords
Robot simulation, path planning, collision detection, distance computation.

1 Introduction

The ability to perform fast path planning is a useful skill for a robot working in a hazardous environment. However, although path planning has been a research field for many years the use of the techniques of path planning are limited outside of the research laboratory. Historically we can identify a small number of reasons for this:

1. *Complexity.* The theory behind path planners is mathematical in nature and can get quite complex. This has important consequences not only when trying to persuade a potential user that the techniques are useful, but also in trying to persuade the user that the program code can be maintained in the field.

2. *Speed.* Path planning has a reputation of being too slow to be useful. (With modern algorithms and modern computers, this need no longer be the case.)

3. *Domain.* Although practical two-dimensional path planners have been available for some time, these are useless in many areas in which complex, three-dimensional geometry is the norm.

4. *Non-generality.* Different path planning programs have different areas of competence, and most fail for many problems. This possibility of failure equates to unreliability in the mind of a potential user.

We have been worrying about these issues in Oxford for some time, and think that we now have a partial solution. We have developed an experimental system for writing path planners called OxSim [12]. The key feature of OxSim is the separation of *geometric competence* from *planning competence*, that makes the writing of path planners within OxSim relatively straightforward. This works by writing the path-planning function in terms of the distance

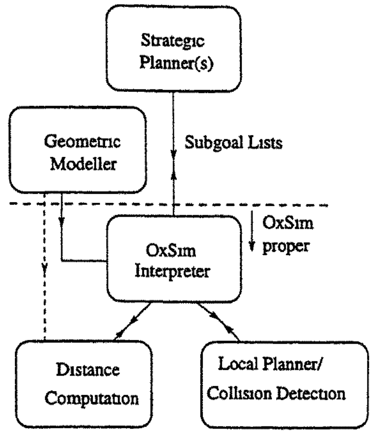


Figure 1: Major Components of OxSim

between the links of a manipulator and the surrounding obstacles. The path planning code does not need to worry about how these distance calculations are implemented, and the writer of the path planner can concentrate on how to implement a planning strategy, rather than on geometry. Although this use of distances may, at first, seem quite restrictive, this is not the case. Most potential field methods can naturally write down their potential fields as functions of distance; techniques based on collision detection can use distance as their test for collision; and most configuration space based techniques are either too global and too slow to be practical, or if they probe subsets of configuration space then these probes can normally be expressed as distance measures.

2 OxSim Overview

A functional overview of OxSim is show in Fig. 1. A geometric modeller is used to produce polyhedral models of parts, and convert them into a suitable input format, and OxSim is compiled with an appropriate local path planner. Then commands are sent to OxSim to, essentially, run a simulation of the situation, guided by the local planner and by explicit calls to the distance computation module.

A key part of OxSim is thus the computation of the (Euclidean) distance between links of the robot and the environment. It is not always necessary for this measurement to be accurate when objects are far apart, but the distance measure will be called many times as part of an inner loop of the planner. We have used a variation on the algorithm of Gilbert, Johnson and Keerthi (GJK) for convex polyhedra [6, 3]. One modification that we have made is to store enough information from one run of GJK in order to speed up the next run; that is, we exploit *temporal coherence* when we can. As robot path planners tend to step gradually forward in time, this single modification has a dramatic effect on computation times. Another modification is to improve the hill-climbing mechanism within the algorithm. Experimental results show that each call to the algorithm can be achieved in around 30-50μs on a 120MHz Pentium if the relative velocities are not excessive [4], which is comparable with the Lin-Canny algorithm [9].

Above this routine for single pairs of objects we require a mechanism to reduce the potential quadratic cost in the number of possible object pairs. This is currently done using a hierarchy of approximations to the shapes, and some heuristic guidelines [12].

3 Using OXSIM

By incorporating different strategic planners and local planners we can use OXSIM in many ways.

3.1 Collision Detection

Strictly speaking one should distinguish between *interference detection*, which checks whether two static objects are overlapping in space, and *collision detection* in which we check whether objects interfere over a whole range of motions. However in practise the most common way of implementing collision detection is by using a number of interference checks [2].

Given a means of computing distances then, interference detection is easily performed by asking whether the distances between objects is positive. For collision detection it then makes sense to choose a number of time values and check the distance at each value. By examining these values in time order we can then take advantage of temporal coherence in the distance checking mechanism, and also often reduce the number of distance checks required, for if a robot has maximum velocity v and is known to be no closer than d to any obstacle at some time t_i, then the distance cannot reach zero before time $t_i + \frac{d}{v}$. Alternatively, if we do not have any velocity information then we can set a lowest level of temporal resolution, and just perform a simple sequence of tests using a uniform time step.

3.2 Dumb Search Path Planning

Given a robot with configuration space $\mathcal{C}$ then a simple method of path planning is as follows. Construct a grid $\mathcal{G}$ over $\mathcal{C}$, and then the potential links between two adjacent configurations in the grid form a dense mathematical graph. By using a collision detection routine we could check (conservatively) whether the neighbours are indeed connected by a direct path, so consider the graph G whose edges are such inter-neighbour paths. Then we can now cast the path planning problem as a standard graph-search problem over G, and use a standard graph-search mechanism, such as $A^{\star}$, to find shortest paths in the graph. This will, in general, only explore a fairly small subset of G, and by only performing the collision checks to establish the presence of particular edges in G as required (i.e., lazily) we can reduce the number of collision checks to a relatively small number.

This mechanism can be implements easily using OXSIM; we used the expert-system shell CLIPS to implement the search mechanism. It is limited in its effectiveness, but when it does work is very fast, as so is worth using as an initial strategy and then moving on to a smarter strategy if it fails.

3.3 Simulated Annealing

The previous planner may fail if the grid is not dense enough to capture all of the necessary 'avenues' in $\mathcal{C}$. Increasing the grid resolution is not the best answer, because one is likely to find that, as there are 'thin' avenues in $\mathcal{C}$, then the search mechanism may have to explore a significant proportion of G. Instead our simulated annealing planner uses the same idea of lazily constructing a graph over $\mathcal{C}$, but uses a probabilistically more powerful searching mechanism [11].

This planner is able to solve quite complex problems, and again uses the OXSIM framework together with a guidance program written using CLIPS. However for simpler problems it takes significantly longer than the dumb planner. Fig. 2 shows an example of this planner in action with a 5 dof mechanical digger in a tight situation. (Unlike most mechanical diggers this has a parallel linkage at link 2.) Planning for this example took about 20s on a SGI Indy 4600.

We have also used *genetic algorithms* as an alternative probabilistic searching mechanism for a particular situation, namely the discovery of optimal path for a welding robot [8]. This time we ignore the distance between the welding torch and the workpiece (as it's zero), and use an alternative mechanism (also based on distance calculations) to check for collisions.

3.4 Local Planning

Rather than using a planner based on the use of configuration space, we can use potential field based on the workspace instead. In the virtual spring method [10] a potential field is set up to attract the links of the robot towards the goal, whilst repelling them from obstacles. Descriptions of potential field algorithms are normally based on explicit formulae for constructing the field in terms of the individual geometric features of objects (i.e., edges and faces), but instead using OxSim we can compute the current value of the repulsion or attraction vector using the instance between objects, and the direction in which that distance occurs. Thus even if the geometry is quite complex, the *planning* routine is not altered; instead, the geometric calculations take a little longer.

Note that such a mechanism is a local planning mechanism only—it only returns a promising direction for the next move. However we have added the *greyhound and the hare* paradigm [7][5, Ch.13] over the virtual springs mechanism, in which a nominal path for the workload is first produced by some higher-level process. This takes the form of a line in the workspace, with the idea being that the local planner will try to move the payload (or whole or part of the arm) so that it is close to the line. The 'hare' then is a moving goal that tracks along the given line from the start position to the goal position, with the robot—the 'greyhound'—following it as closely as possible.

The line might well be presented to the system by having a human user sketch it; we see this as a powerful and natural way for a human operator to become involved in the planning process, without having to get involved in the low level details. Alternatively, some other planning program might be used, such as the other planners mentioned here. Then the higher level planner might be used at a low level of resolution, and the path is then refined by the local planner.

4 Other Planners

We believe that other motion planning algorithms may be emulated using OxSim. To illustrate, we will briefly consider two famous planners.

4.1 Probabilistic Roadmap Planner

Kavraki's Probabilistic Roadmap Planner (PRM) is deceptively simple. Like the planners of §3.2 and §3.3 it constructs a graph G of possible routes, but unlike those planners it does not use a regular grid and does not perform lazy evaluation of $\mathcal{C}$. Instead it starts by choosing a large number of random configuration, and keeps those which are in free space (i.e., for which an interference check returns false). These configurations will form the vertices of a graph. Then it attempts to find local paths between nearby vertices, and if so adds a graph edge. Thus it computes the graph that will be searched explicitly, which is time-consuming for complex three-dimensional scenes. However it has the great advantage that, for static scenes, paths can then be found in very short times.

Emulating PRM under OxSim should be simple, as the graph is constructed using collision detection and distance information. (The latter is required for a graph refinement step that is not detailed here.) However the random nature of the chosen configuration points means that the temporal coherence of the distance mechanism would not be activated, which means that each interference check might take about 3 times as long as it would otherwise. This effect could be diminished by storing the coherence information—a few tens of bytes—at a set of configuration vertices as the graph is constructed, so that the cached information could be retrieved (via a b-tree structure, or similar) and used to seed the distance mechanism.

4.2 Randomised Path Planner

The Randomised Path Planner (RPP) is probably the most famous path planner, despite having been conceived over 6 years ago. It was the first to demonstrate practically fast planning for

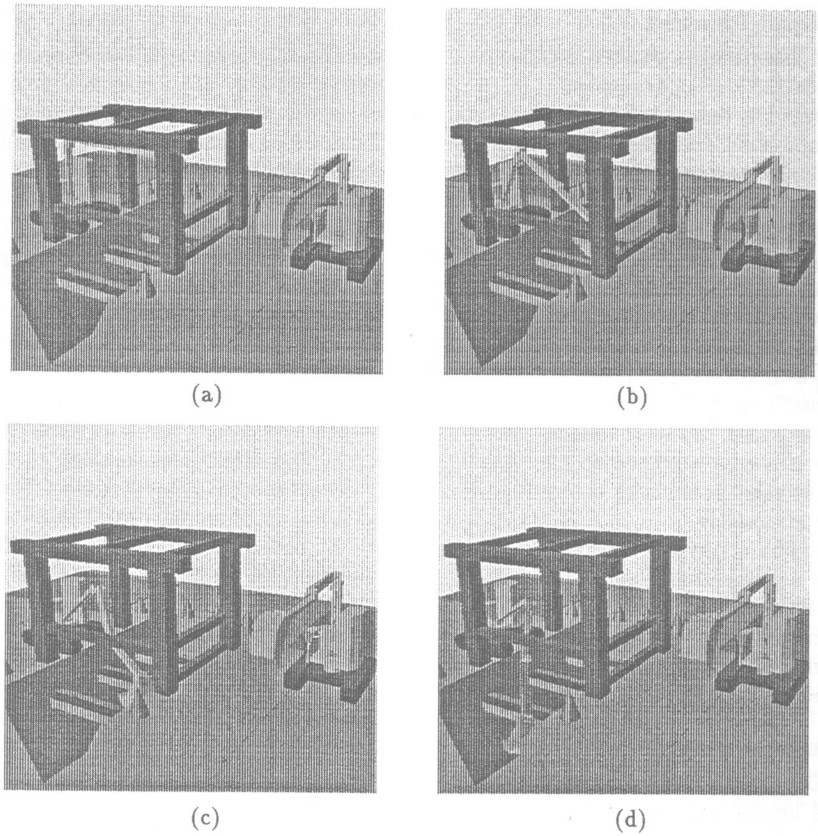


Figure 2: OxSIM in action

kinematically complex manipulators [1], and helped to inspire our own work on the virtual springs planner. Like the virtual springs planner it uses a potential field in the workspace of the robot as a local planning mechanism, together with a higher-level planner that is based on a search of $\mathcal{G}$, and a random walk mechanism to try to escape from local minima.

Exactly emulating RPP using OxSIM is difficult, as RPP uses a grid in the workspace, and computes distances on that grid using a wavefront mechanism. These distances are then used both for high-level planning, and also to define the low-level potential field used. However it should be possible to use a purely distance-based potential field for the low-level planner, as in the virtual springs planner, and then to generate the graph for the high level planner lazily.

5 Conclusions

In this paper we have expounded on the principles of *problem decomposition* and *the separation of geometry and planning* that allow (we claim) the development of planners that are useful outside of a research environment. Our current scheme makes use of the enhanced version of Gilbert, Johnson and Keerthi's minimum distance algorithm, together with the virtual spring method, and works well for quite complicated situations in 3D environments. Instead of using stick-like links, we replace the links with real geometries and represent the links in a hierarchical data structure to attain efficiency.

We have also experimented with several alternative planners, and now are re-engineering OxSim to enable us to try emulating other planners. If such emulation proves viable then we claim that an approach like OxSim will make it much easier to prototype planners, and the new code in the local and strategic planners is comparatively small (100-1000 lines of code so far). It may even make sense to produce final implementations this way, as the gains in speed to be had by producing an optimised implementation may well be small compared with the additional costs associated with the added software maintenance. (Geometric code, such as we have developed for distance calculations, is very hard to write and change.)

We have also started a new project in Oxford, partly in order to test whether the OxSim philosophy will scale to make solutions of industrial-sized problems possible (and easy to maintain). This project will deal with the planning of robot motions within large and slightly flexible structures such as aircraft wings. The geometric descriptions of these structures are large (tens to hundreds of thousands of polygons) and definitely curved, and provides a considerable challenge to our ideas.

Acknowledgement

This work has been financially supported by the EPSRC.

References

[1] J. Barraquand and J.C. Latombe. Robot motion planning: A distributed representation approach. *Int. J. Robotics Res.*, 10(6):628–649, 1991.

[2] S. A. Cameron. A study of the clash detection problem in robotics. In *Int. Conf. Robotics & Automation*, pages 488–493, St. Louis, March 1985.

[3] Stephen Cameron. A comparison of two fast algorithms for computing the distance between convex polyhedra. Conditionally accepted, IEEE TR&A, July 1996.

[4] Stephen Cameron. Enhancing GJK: computing minimum and penetration distances between convex polyhedra. In *Int. Conf. Robotics & Automation*, April 1997.

[5] Stephen Cameron and Penny Probert, editors. *Advanced Guided Vehicles: Aspects of the Oxford AGV Project.* World Scientific Press, 1993. In preparation.

[6] E. G. Gilbert, D. W. Johnson, and S. S. Keerthi. A fast procedure for computing the distance between complex objects in three-dimensional space. *IEEE Trans. Robotics & Automation*, 4(2):193–203, April 1988.

[7] T. Hague and S. Cameron. Motion planning for non-holonomic industrial robot vehicles. In *Proc. IROS-91*, pages 1275–1280, Osaka, November 1991. IEEE.

[8] Masato Kageyama and Stephen Cameron. Rapid path planning for an industrial welding robot. In preparation, 1997.

[9] Ming Lin and John Canny. A fast algorithm for incremental distance calculation. In *Int. Conf. Robotics & Automation*, pages 1008–1014, Sacremento, April 1991.

[10] A. W. McLean and S. A. Cameron. The virtual springs method: Path planning and collision avoidance for redundant manipulators. *Int. J. Robotics Res.*, 15(4):300–319, August 1996.

[11] Caigong Qin. *Path Planning for Complex Manipulators.* PhD thesis, Computing Laboratory, University of Oxford, 1996.

[12] Caigong Qin, Stephen Cameron, and Alistair McLean. Towards efficient motion planning for manipulators with complex geometry. In *Proc. IEEE Int. Symp. Assembly and Task Planning*, pages 207–212, August 1995.

DESIGN/MANUFACTURE CNC MILLING MACHINE WITH OPEN STRUCTURE IN CONTROL SYSTEM

M.J. Wu, M.H. Shaw, C.W. Lee and C.C. Wang
Department of Industrial Education
Chang-Hua University of Education
Chang-Hua, Taiwan, R.O.C.

ABSTRACT

The control system of CNC machine in the industry is in a close form, and its price is expensive. In this article a CNC milling machine adopts National FP-3 PLC as the control system. NATIONAL FP3 PLC is a modular controller on which Central Processing Unit(CPU), Position Control Unit with two axes and one axis(PCU), I/O Unit, D/A converter Unit, and Computer communication Unit(CCU) mounted. The personal computer with 486 CPU linking with CCU is to transmit data into PLC and a station of LAN. The program will convert G-codes of the cutting path into numerical data suitable for PLC program, perform data communication with National FP-3 PLC through CCU modular. In this CNC milling machine, the systems of mechanic frame are an x-y table driven by PANASONIC servo-motors and a cutting axis that have AC 220V conduct motor and servo motor. An example of cutting a cube through the LAN is presented to test system operation, and the result of motion of CNC milling machine is correct.
Keyword: PLC, manufacture, network, control system.

1. INTRODUCTION

Since the digital computer has been created, people started to apply it to modify the automatic manufacture processes for more than 30 years. In the computer numerical control (CNC) of machine, controlling programs could be stored in memory and did not have to be reloaded each time a part was to be produced. Also, certain "canned" routines could be used to accomplish many repetitive functions such as peck drilling and tapping. In CAD/CAM system, the input for CNC machines is an another computer in which has a CAM software such as SmartCAM to generate G-codes of object. When G-codes has been transmitted, the control and arithmetic elements use the program to drive the tooling through the desired motion.

The programmable logic controller (PLC) was developed in the early 1970s for General Motor. In the industry, it already has been applied widely in control of manufacturing processes, assembly systems, and general automation control. With different function module's combination, PLC performances become variety. In this article, a CNC milling machine with FP3 PLC and modules as its control system is designed and manufactured in the laboratory to simulate the functions of that used in the industry.

In FMS system, every machine must be linked to form a network that is called a LAN (Local Area Network). In linear bus network such as Ethernet all stations are connected with a single length of cable (called a bus). All stations on the network can transmit a message to any other station on that network; consequently, each station must inspect the address of all messages.

After completing work, the control system is linked with a local area network successfully. The LAN server is Microsoft NT 3.51 and the operation system of the station is Microsoft Win95. The following sections will describe the whole system. An example of cutting a cube through the LAN is presented to test system operation, and the result of motion of CNC milling machine is correct.

2. HARDWARE OF THE MACHINE

Since a CNC milling machine can perform sophisticated movement, it is used to manufacture most of mechanic parts in the industry. Generally, a CNC mechanic structure contents X, Y, Z axis and cutting-driving motor. X and Y axis forms an X-Y table on which mechanic part is mounted. The cutting-driving motor and Z axis are fixed together to become a cutting system. In this project aluminum-extrusion beam and steel plate construct the mechanic frame on which four axes are fixed. The following will describe the mechanic structure of X-Y Table and Cutting System.

X-Y Table: The mechanic structure of X-Y Table consists of ball screws and a pair of parallel linear guides on x and y direction. The ball screw will move X-Y Table forwards and backwards on x or y direction. Each pair of linear guides will make the table move linearly without rotation. The driver system of the table is PANASONIC sever-motors.

Cutting System: The mechanic structure of the cutting system consists of a 220ACV conduct motor and one direction moving table on the z axis which driver is a PANASONIC server-motor. The motor speed is controlled by a IGBT inverter. The table makes up of a ball screw and a linear guide.

3. CONTROL SYSTEM OF THE MACHINE

Since the control system of CNC machine in the industry is packed in a closed form, people are very hard to understand the organization of the control system. Here a CNC milling machine, shown in fig. 1, with NATIONAL FP3 PLC, its modules as the controller, shown in fig. 2, and a personal computer will simulate the function of that in the industry. Each part of the control system will be described as following.

- FP3 PLC: The NATIONAL FP3 PLC [1] is a modular PLC with a rack on which modules are mounted.
- Power Supply Unit: Since the voltage of the control system is 24 DCV, PSU function is to convert 110 ACV to 24 DCV stable.
- Central Processing Unit (CPU): The computer, called the processor, is at heart of the PLC operation. It will command the whole processes according to the ladder diagram of PLC.
- Position Control Unit: The function of PCU module [2] is to control the speed and rotation angle of the server motor that links with the ball screw to move tables precisely. PCU for X-Y table is a two-ax's module that has the function to make the table move in circle. PCU for Z table is a one axis module.
- Input Unit & Output Unit: The performance of I/O module is to receive the signal from Origin Position sensor, limit position sensor, start button signal, and the power switch of cutting motor.
- D/A Unit: According to material property, the speed of the cutting-motor of CNC milling machine will be changeable. The function of D/A Unit [3] will receive the data from CPU and convert that into analog signal to control IGBT inverter.
- Computer Communication Unit: The function of CCU module [4] is to receive or transmit the data between FP3 PLC and the personal computer.
- IGBT inverter: IGBT links with the cutting motor and D/A module. According the signal from D/A Unit it will start to vary the frequency of the motor that means change the speed.

4. APPLICATION SOFTWARE

The software of the CNC milling machine includes SmartCAM, Visual Basic and NPST for ladder diagram editor. The performance of each software will be stated as following.

- SmartCAM Advance 3 D for milling: In the industry SmartCAM [5] is one of software packages for CAD/CAM. The main function of the package are to creating 3 D graphics, simulate the manufacturing processes, convert 3 D graphics to G-codes
- Visual Basic: The designer use VB to program that will convert G-codes to the data that suitable for FP3 PLC memory requirement. Another VB program will communicate with LAN and FP3 PLC.
- NPST: NPST is ladder diagram editor. The designer use NPST to design the ladder diagram of control system.

Before the PLC program, shown in fig. 3, design starts, parameter tables of modules require set up first. Tables include parameters setting, external and internal I/O schedule, CPU memory map, and communication protocol, shown in table 1,2,3,4.

5. FLOWCHART OF CONTROL SYSTEM OPERATION

An example of cutting process of cubic object has been demonstrated successfully by this open structure control system linking with a LAN. The cutting processes will be stated as following

- CAD/CAM: The designer, using SmartCAM, draws 3D object, simulates manufacturing processes, converts the graphics into G-codes, and saves that in the hard disk, shown in fig 4.
- G-codes conversion. The program, written in VB, reads G-codes file, converts each G-codes into data suitable for FP3 PLC program and saves as a new file
- LAN communication: In this project, the computer of the G-codes conversion data file, named as "A," links LAN server and the computer of control system of CNC machine through network adapter and coaxial cable The operator transfers the data file from "A" computer to the computer of control system of CNC milling machine.
- Data transmitting. The operator begins to execute the program in the computer of the control system from "A" computer directly. Then the computer of the control system starts to transmit the data to CCU module through RS 232 PLC program will move the data in CCU to Data Memory of CPU.
- Movement of machine PLC program starts to catch the data from the memory in sequence and execute the relative subroutine that controls the operation of PCU modules and D/A Unit According data from subroutine, PCU starts to send pulses into the driver of servo-motor that will move the X-Y table and Z table.
- Signal Feedback: During PLC program design, Data Memory of CPU is separated into two sections. When the anyone section of data has been executed, a signal will be send back to the linking computer, and a section of data will start to transmit to that area After all the data execution completing, PLC motion will stop and the terminal computer with operator will receive signal of end.

6. CONCLUSION

Since the cutting steps are correct, to replace close form of control system of CNC milling machine with open structure PLC controller is possible. In the future Ethernet communication module replacing CCU will make the control system linking with LAN more simple and direct. The data can be transmitted directly from the terminal of LAN without a computer linking with CCU.

7. REFERENCES

1. FP3/FP5 PLC technical Manual, Matsushita Electric Works, Ltd.
2. FP3/FP5 Position Control Technical Manual, Matsushita Electric Works, Ltd.
3. FP3/FP5 D/A Converter Technical Manual, Matsushita Electric Works, Ltd.
4. FP SERIES C.C.U (Computer Communication Unit) Technical Manual, Matsushita Electric Works, Ltd.
5. SmartCAM Advanced 3-D Machining Reference Manual, Point Control Co., 1994.

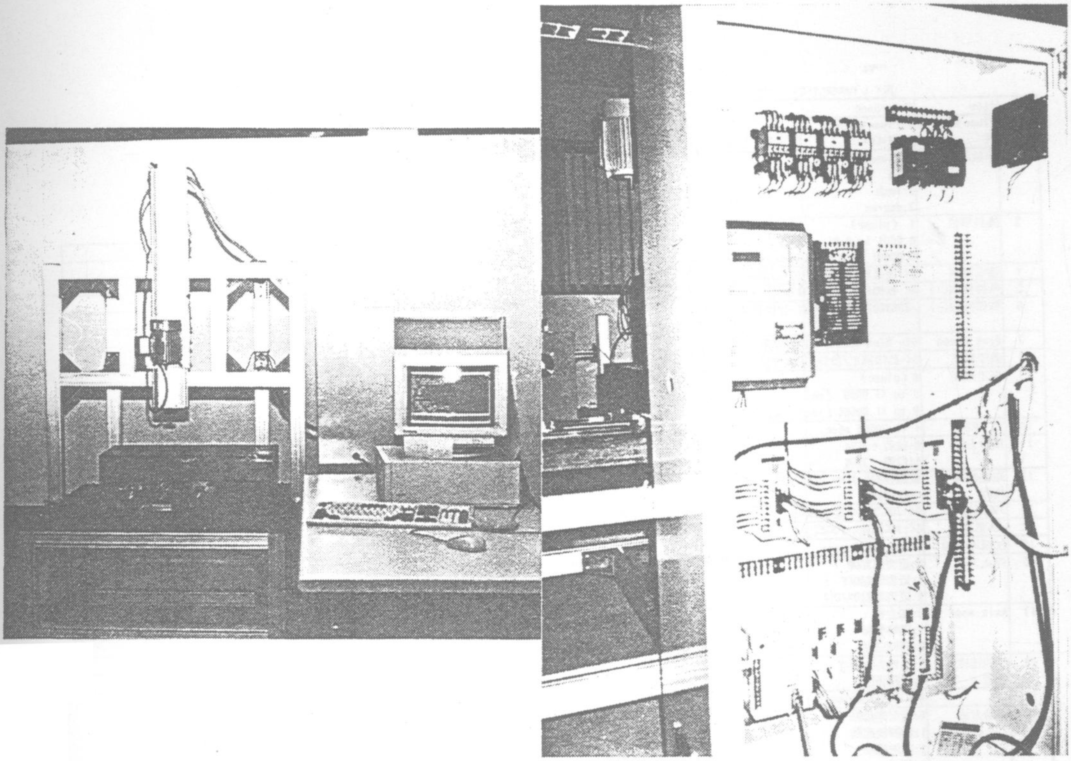


Figure 1: CNC milling machine in laboratory

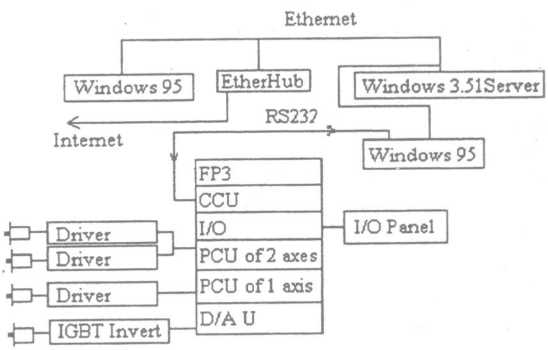


Figure 2: System diagram of control system

表4 1 PARAMETERS 功能設定

NO	Item	Set range	Default Value
1	Pluse 輸出模式	0:Pluse+Sign 1:CW+CCW	1
2	單位設定	0:Pluse 1:mm 2:inch 3:degree	0
3	換算單位	1 (pluse) 0.0001 to 0.01 (mm) 0.00001 to 0.001 (inch・degree)	1
4	速度極限	0<-速度極限/換算單位<-400000	400000PLS/S
5	軟體極限(+)	0<-軟體極限(+)/換算單位<-8388607	8388607PLS/S
6	軟體極限(-)	8388607<-軟體極限(-)/換算單位<-0	- 8388607PLS/S
7	Bias speed	0<= Bias speed <=速度極限	0 PLS/S
8	齒隙補正	0<=齒隙補正/換算單位<=255	0 PLS/S
9	誤差補正	0 (pluse) 0 to 11.0000 (mm) 0 to 11.00000 (inch・degree)	0 PLS/S
10	完了時間	1 to 20000 msec.	300 msec.
11	復歸方向	0:位置+方向 1:位置-方向	1
12	原點復歸位置	軟體極限(-) <= 復歸位置 <-軟體極限	0 PLS/S
13	JOG高速	0<=JOG高速<=速度極限	50000 PLS/S
14	JOG低速	0<-JOG低速<=JOG高速	100 PLS/S
15	加減速時間	64 to 49999 msec.	1000 msec.
16	停止方法	0:近旁原點ON 1:近旁原點OFF 2:近旁原點ON/OFF	0
17	Axis mode	0:獨立軸 1:同時雙軸 2:同時三軸	0
18	補間速度指定	0:長軸方向速度 1:追蹤速度	1
19	I/F	Bit designation(0.1)	All 0
20	起動方法	0:通常即起動 1:復歸後起動 2:高速起動 3:測試	0

Table 1: Parameters setting

表4 5 P.C.U. Share Memoty Map

31FH - 3F0H	系統OS範圍	
32AH	Error code (Z軸)	(R/W)
329H	Error code (Y軸)	(R/W)
328H	Error code (X軸)	(R/W)
324H	現在位置 (Z軸)	(R)
322H	現在位置 (Y軸)	(R)
320H	現在位置 (X軸)	(R)
31DH	現在位置變更 (Z軸)	(W)
31CH	現在位置變更 (Y軸)	(W)
319H	現在位置變更 (X軸)	(W)
318H	傳送變更NO.	(W)
314H	JOG速度 (Z軸)	(R/W)
312H	JOG速度 (Y軸)	(R/W)
310H	JOG速度 (X軸)	(R/W)
30AH	補助出力 (Z軸)	(R)
309H	補助出力 (Y軸)	(R)
308H	補助出力 (X軸)	(R)
302H	起動NO. (JOB3)	(W)
301H	起動NO. (JOB2)	(W)
300H	起動NO. (JOB1)	(W)
280H	Parameter (Z軸)	(R/W)
200H	Data (Z軸)	(R/W)
180H	Parameter (Y軸)	(R/W)
100H	Data (Y軸)	(R/W)
080H	Parameter (X軸)	(R/W)
000H	Data (X軸)	(R/W)

Table 2: CPU memory map

表4-3 P.C.U. I/O接點配置表

X	P.C.U.-->PC	Y	PC-->P.C.U.
X0	P.C.U. ready	Y20	Sequencer ready
X1	錯誤檢出	Y21	讀出傳送
X2	Run(接點OFF)/Local(接點ON)	Y22	寫入傳送
X3	讀出傳送完了	Y23	JOB1始動
X4	寫入傳送完了	Y24	X軸機械原點復歸始動
X5	JOB1 Positioning 完了	Y25	X軸軟體原點復歸始動
X6	X軸原點復歸完了	Y26	JOB1停止
X7	JOB1 BUSY	Y27	X軸正轉JOG
X8	JOB1始動完了	Y28	X軸逆轉JOG
X9	JOB1補助出力ON	Y29	JOB1補助出力OFF
XA	JOB2 Positioning 完了	Y2A	JOB2始動
XB	Y原點復歸完了	Y2B	Y軸機械原點復歸始動
XC	JOB2 BUSY	Y2C	Y軸軟體原點復歸始動
XD	JOB2始動完了	Y2D	JOB2停止
XE	JOB2補助出力ON	Y2E	Y軸正轉JOG
XF		Y2F	Y軸逆轉JOG
X10		Y30	JOB2補助出力OFF

Table 3: External and internal I/O schedule

Transferblock NO.	Data contents
0	Parameters (X.Y.Z)
1	Data nos.1 to 10 (X.Y.Z)
2	11 to 20
⋮	⋮
39	381 to 390
40	391 to 400
41	X軸現在位置
42	Y軸現在位置
43	Y.X現在位置
44	Z軸現在位置
45	Z.X現在位置
46	Z.Y現在位置
47	Z.Y.X現在位置

Table 4: Communication protocol

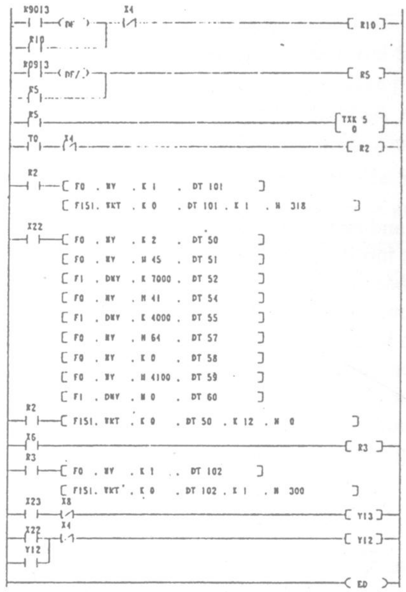


Figure 3: Part of FP3 PLC program

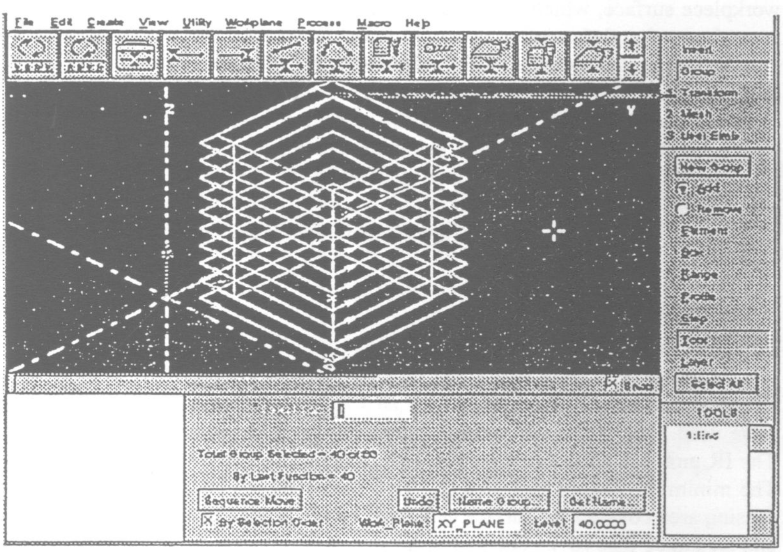


Figure 4: Cutting processes in SmartCAM

INVESTIGATION ON MACHINED SURFACE TEMPERATURE IN TURNING

T.H. Chu and J. Wallbank
Department of Engineering
The University of Warwick
England, U.K.

ABSTRACT

A computational and experimental method for the determination of the temperature close to the primary cutting edge in turning has been developed. The cutting temperatures of 0.1%C low carbon free cutting steel have been measured for a range of cutting speeds and feedrates at a constant depth of cut. The results show that the temperature correlates well with cutting speed and feedrate. The cutting forces have also been related to the deformation of the work material near the cutting edge of the tool. The calculation of the workpiece temperature distribution using a finite element method has been carried out with the aid of the measurement of forces. It is suggested that the predictive models developed by this method can be incorporated into a computer aid manufacturing system to obtain improved control during cutting

KEYWORDS

Metal-cutting, Cutting-temperature, Cutting-force, Finite-element-method, Infrared-sensor

1. INTRODUCTION

Many experimental measurements [1,2] indicate that it needs more energy for the actual cutting operation than that predicted by Merchants model [3]. This means that a proportion of the total force is not accounted for and it has been proposed that this is acting between the tool flank and the workpiece surface, which does not contribute directly to chip formation [4]. Research results [5, 6] have shown that deformation occurs in a narrow primary deformation zone and that the forces in the zone are supported by the material surrounding it. A stress field is built up and induces elastic-plastic deformation within the area that is under the machined surface and near the cutting edge. Wallbank observed that, even with a sharp tool, the workpiece material is in contact with the clearance face for a distance of the order of 0.2 mm below the cutting edge of a tool with a clearance angle of about 6° [7]. In this study, temperature measurement of the newly created workpiece surface using a commercial infrared sensor during turning a low carbon free cutting steel has been carried out. The objective is to be able to predict these temperatures allowing better dimensional control in turning.

2. METHOD

2.1 Experimental procedure

Cutting tests were performed on a 25 kilowatt CNC controlled lathe under dry conditions. An uncoated carbide (P40) insert having an ISO designation SNMG-12-04-04 with a zero rake angle were used. The machined surface temperature was measured with an infrared non-contact sensor. The IR unit is a close focus sensor with a detecting element using the 8-14 micron spectral region. The minimum sample spot size (2.5 mm diameter) is obtained at a focal distance of 76 mm. The sensing area covered the surface generated by the primary cutting edge through the gap created by the end clearance face (6°). The tool insert mounted on a tool holder gave a 45° approach angle. Use of a 45° approach angle ensured that the 2.5 mm sensing area was fully accommodated by a 2 mm depth of cut. A three-component dynamometer was used for measuring cutting forces in turning. A constant depth of cut of 2 mm was used with cutting speeds varying from 100 M min^{-1} to 200 M min^{-1} and feedrates of 0.05 $mm\ rev^{-1}$, 0.1 $mm\ rev^{-1}$ and 0.15 $mm\ rev^{-1}$.

2.2 Temperature modelling

When there is a small area of contact between the tool flank face and freshly machined surface, the orthogonal cutting model of a sharp tool can be modified to include the plastic deformation which can be considered as a heat source (Fig.1a). Flow near the cutting edge, A, is observed in: 1. the primary shear zone, 2 the secondary shear zone and 3. the tertiary shear zone, where F_c and F_f are force components in the cutting and feeding directions respectively, lc is the length of contact between the work material and the flank face of the tool with a clearance angle δ, and F_{cr} and F_{fr} are the forces acting on the flank face in cutting and feeding directions respectively (Fig.1b). The heat flowing into the workpiece from the primary zone and the tertiary zone are two main heat sources which cause the temperature increase in the work material.

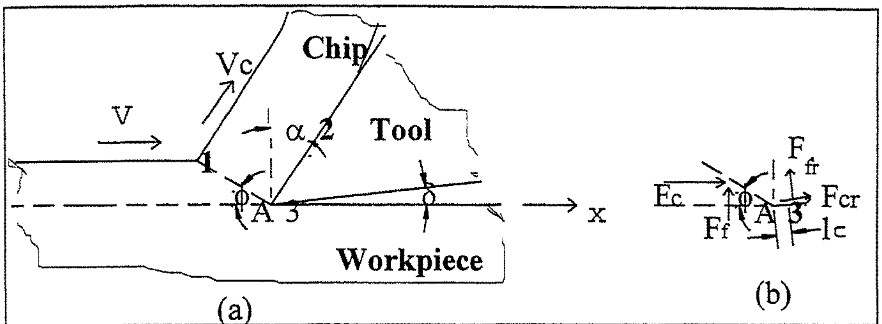


Fig.1 Orthogonal cutting model (a). heat sources and (b). force diagram.

2.2.1 Heat in chip formation

The heat generated on the shear plane (when the rake angle, α, is zero) is

$$Q_s = F_s V_s = \frac{(F_c \cos\phi - F_f \sin\phi\ )\ V}{\cos\phi} \qquad (1)$$

where ϕ is the shear plane angle. Most of the heat generated on the shear plane passes into the chip, but a proportion is conducted into the workpiece material. Many methods [8, 9] have been applied to determine the fraction (ß) of shear plane heat carried away by the workpiece. Trigger and Chao [10] and Trent [11]assumed that ß was 10%. Then, the heat per unit time per unit area that flows into the workpiece is

$$q_1 = \frac{\beta\ (F_c \cos\phi\ - F_f \sin\phi\ )V \sin\phi}{w t_1 \cos\phi} \qquad (2)$$

2.2.2 Heat flow at the interface between the tool flank face and the machined surface

Assuming that the heat flux along the interface between the tool flank face and the machined surface is uniform, the heat flux due to the heat generated at this interface can be estimated as

$$q_W = \frac{Q_W}{l_c w} = \frac{F_{cr} V}{l_c w} \qquad (3)$$

where F_{cr} is the residual force acting on the flank face in the cutting direction and l_c is the length of contact between the tool face and machined surface.

2.3 The finite element simulation

The application of a finite element method for calculating the temperature distribution in metal cutting has been described by Tay et al. [12] and Childs et al. [13]. The energy equation subject to the boundary conditions of the problem is described as below (Fig.2):

(1). The governing equation :

$$\frac{k}{\rho c}\left[\frac{\partial^2 T}{\partial x^2}+\frac{\partial^2 T}{\partial y^2}\right]=u_x\frac{\partial T}{\partial x}+u_y\frac{\partial T}{\partial y} \qquad (4)$$

where u, k, ρ, and c are velocities of material flow, thermal conductivity, density, and specific heat, respectively. (2). The boundary conditions: specified boundary temperature To on surface S_T, specified heat flux due to the heat source (q_1, q_w) on surface Sq, and heat transfer: $q_h = h(T - T_0)$ on surface S_h, where h, T and To are the coefficients of heat transfer, temperature of the workpiece or tool and temperature of the air, respectively.

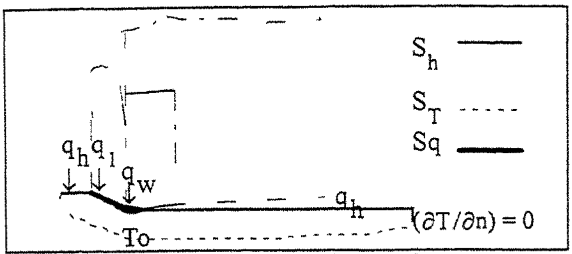


Fig.2 Boundary conditions of the cutting model

3. RESULTS

The temperature of the workpiece surface at 3, 6 and 9 mm below the cutting edge was measured. To estimate the temperature at the cutting edge exponential extrapolation was used. The results of this at 120 seconds are shown in table 1. Flank wear measured after each test is less than 0.2 mm.

Table 1: Temperature prediction for the cutting edge

V	f	Temperature from the cutting edge			Predicted temperature
(M/min)	(mm/rev)	3 mm	6 mm	9 mm	near cutting edge
100	0.05	87	74	63	102
100	0.1	95	80	67	113
100	0.15	114	88	68	147
125	0.05	93	77	64	112
125	0.1	104	84	68	129
125	0.15	118	95	77	146
150	0.05	99	84	70	118
150	0.1	111	89	71	138
150	0.15	126	102	82	156
175	0.05	104	88	66	120
175	0.1	114	93	76	140
175	0.15	127	103	83	157
200	0.05	110	88	68	137
200	0.1	124	100	81	153
200	0.15	132	110	88	173

The measured forces are a summation of those on the primary shear plane, the secondary shear plane and the contact on the flank face. As the clearance angle is small to a first approximation, the cutting force at zero feedrate can be taken as the shear force on the flank face. Then, to separate the first two (the modified cutting forces) from the last one (the flank force), the force is measured at different feedrates and then extrapolated to zero feedrate (Fig.3). A very high value over 0.99 of the correlation coefficient for the regression models was found.

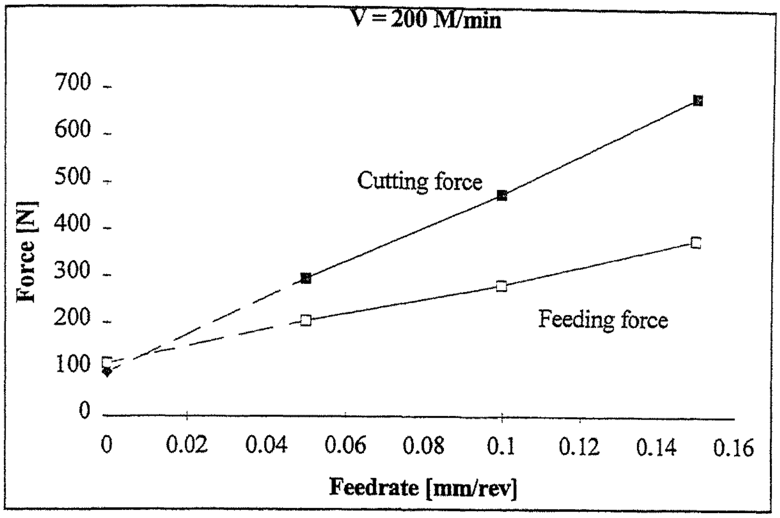


Fig.3 Force extrapolation

The cutting conditions used in the calculation to obtain the temperature in dry cutting are shown in table 2. The cutting forces and shear angles are experimental results. The length of contact between the tool flank face and the machined surface is assumed as 0.2 mm for the sharp tool insert [7]. The calculated results are shown in Fig. 4.

Table 2. Cutting conditions for FEM calculation

Cutting speed (M / min.)- V	200
Feedrate (mm / rev.)- f	0.1
Depth of cut (mm)- w	2
Modified cutting force (N)- F_{cm}	381
Modified feeding force (N)- F_{cf}	168
Force on flank face in cutting direction (N)- F_{cr}	95
Force on flank face in feed direction (N)- F_{fr}	114
Shear angle (°)- ϕ	26
Tool rake angle (°)- α	0
Tool clearance angle (°)- δ	6
Room temperature (°C)- To	20
Work material thermal conductivity(W / m °K)-k	54
Work material density (kg / m³)- ρ	7820
Work material specific heat (J / kg °K)- Cp	460

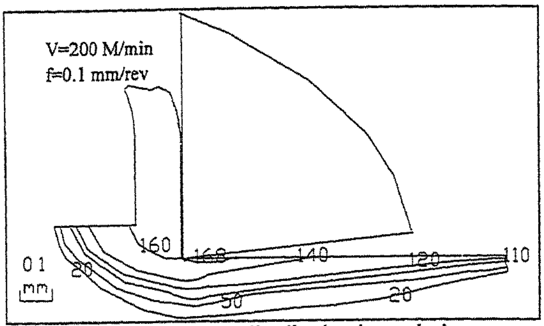


Fig.4 Temperature distribution in workpiece

4. DISCUSSION

Using regression analysis, a relationship between workpiece temperature T and cutting variables (speed V and feedrate f) has the form:

$$T = 46.5V^{0.33}f^{0.25} \tag{5}$$

A strong correlation between temperature, speed and feedrate can be observed. A high value of the correlation coefficient (0.94) shows that the equation should be capable of predicting the workpiece temperature over the ranges tested with a great deal of accuracy.

Fig.4 shows that the maximum calculated temperature (168° C) on the machined surface is obtained at the middle of the contact area between the tool flank face and the machined surface. This is higher than the extrapolated measured temperature as shown in Fig.5. The maximum temperature of the primary zone, 160° C, is slightly lower than the maximum temperature on the machined surface. Fig.5 shows that the temperature distribution curves of the experimental and computational results have a similar trend with distance from the cutting edge on the machined surface. The heat transfer from the machined surface to air is hard to estimate because of the lack of the measurements of heat transfer coefficient near the cutting edge, when the machined surface is moving with the cutting speed. The temperature difference between the measured data and the analytical results on these two approximately exponential curves is within a reasonable range (about 15° C), but it should be noted that this is a very small value given the constants assumed in the analytical model. In ref.13, Childs, Maekawa and Maulik suggested that a lower estimate for the heat flux by conduction through air from the hot tool to the cooler machined surface in the gap between the tool flank face to the workpiece surface is

$$q_{min} = k_a(T_t - T_W)/(x\tan\delta) \tag{6}$$

where k_a is the thermal conductivity of dry air and T_t and T_W are the temperatures of surfaces of the tool and workpiece at a distance x from the cutting edge. If this heat transfer is added into the cutting model (let $k_a = 0.03\ WM^{-2}K^{-1}$, $\delta = 6°$ and $T_t = 500°C$), then the temperature difference between the experimental and computational results remains within 5% as shown in Fig.5.

These results suggest that the surface temperature can be accurately estimated either from extrapolated measured data which can form the basis of mathematical correlations or via heat flow calculations once the forces on the flank face are known. This should allow the dimensional change (due to thermal expansion) to be controlled during machining.

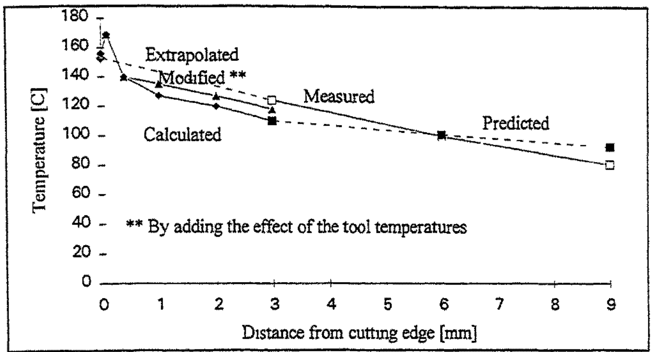


Fig.5 Comparison of computational and experimental machined surface temperature

5. CONCLUSION

An experimental and computational method for predicting the temperature of the newly generated surface of workpiece using a non-contact sensor and the finite element analysis has been developed.

(1) Equations predicting the effects of cutting speed and feedrate have been developed and show good correlation.

(2) Forces acting on the flank face have been shown to be the most significant contributor to the temperatures in the workpiece.

(3) Comparisons between computational and experimental results show good agreement.

6. REFERENCE

1. Albrecht P., 'New developments in the theory of the metal-cutting process', Tran. ASME J. Ind., Vol.82, pp.348-358, 1960.
2. Stevenson R. and Stephenson D. A., 'The mechanical behavior of zinc during machining', Tran. ASME J. Ind., Vol.117, p.172, 1995.
3. Merchant, M.E., ' Mechanics of the Metal Cutting Process. I. Orthogonal Cutting and a Type 2 Chip', J. of Applied Physics, 16, 5, pp.267-275, 1945
4. Williams, J.E., 'Some Aspects of a Three-Zone Model of Machining', Ware, 48 (1), pp.55-71, 1978.
5. Lin, Z.C., Lin, Y.Y. and Liu, C.R., 'Effect of Thermal Load and Mechanical Load on the Residual Stress of a Machined Workpiece', Int. J. Mech. Sci., 33 , 4, pp.263-278, 1991.
6. Wu, D.W. and Matsumoto, Y., ' The Effect of Hardness on Residual Stresses in Orthogonal Machining of AISI 4340 Steel', Tran. of the ASME J. of Eng. for Industry, 112, pp.245-252, 1990.
7. Wallbank, J., 'Surfaces Generated In Single-Point Turning', Wear, 56, pp.391-407, 1979.
8. Weiner J. H., 'Shear-plane temperature distribution in orthogonal cutting', Tran. ASME, Vol.77, p.1331-1341, 1955.
9. Loewen E. G. and Shaw M. C., 'On the analysis of cutting tool temperatures', Tran. ASME, Vol.76, pp.217-231, 1954.
10. Trigger K. J. and Chao B. T., 'Cutting temperature and metal-cutting phenomena', Tran. ASME, Vol.73, pp.777-793, 1951.
11. Trent E. M., Metal Cutting, 3rd Ed., p.58, Butterworth Heinemann Ltd., Oxford, 1991.
12. Tay, A.O., Stevenson, M.G. and de Vahl Davis, G., ' Using The Finite Element Method To Determine Temperature Distributions in Orthogonal Machining', Proc. Instn. Mech. Engrs., 188, 55, pp.627-638, 1974.
13. Childs, T.H.C., Maekawa, K. and Maulik, P., 'Effects Coolant on Temperature Distribution in Metal Machining', Materials Science and Technology, 4, pp.1006-1019, 1988.

FORECASTING AND CONTROL OF THE COMPONENT ACCURACY IN TURNING WITH ADVANCED ALGORITHMS

K. Cheng, D. Kirkwood and X.H. Chen
Department of Engineering, Glasgow Caledonian University
Cowcaddens Road, Glasgow G4 0BA
United Kingdom

ABSTRACT

This paper proposes an approach to achieving better machining precision through errors forecasting and control with algorithms during the machining processes. The algorithms include autoregressive (AR) and autoregressive moving average (ARMA) models and genetic algorithms An application example is presented to demonstrate the approach used in CNC turning process Simulations using LabVIEW are developed to verify the proposed approach.

KEYWORDS

Forecasting and Control, Machining Accuracy, Time Series, Genetic Algorithms

1. INTRODUCTION

The accuracy of the workpiece is defined as the degree of conformance of the finished part to dimensional and geometric specifications[1]. The performance of machine tools in terms of accuracy is determined by the error of the relative movement between the cutting tool and the ideal component. The approaches to achieving high machining precision are all focused on reducing the error of the relative movement even though they use the principle of 'error avoidance' or 'error compensation.'

The cost of error avoidance techniques increases exponentially in the high accuracy range and imposes economic constraints on their practical applicability. High precision machining depends on not only the higher accuracy machine tools but also precise control of the varying machining conditions which are normally difficult and expensive. With the advances in computer based control and sensing technologies, the implementation of the 'error compensation' approach becomes possible and practical.

In this paper the 'error compensation' approach is explored to achieving precision machining based on algorithm-based forecasting control. Time series algorithms and genetic algorithms are presented to illustrate their usage in predicting and control of the machining errors in CNC turning. The simulated results are achieved using LabVIEW programs to verify the proposed approach.

2. MACHINING ACCURACY AND FORECASTING CONTROL

2.1 Proposed the System Architecture

There are many factors affecting the machining process and thus the machining accuracy. The factors include:

- the geometric and kinematic features of the machine tools,
- the cutting tool geometry, setting and wear,
- the varying environment such as temperature and vibrations,
- the workpiece characteristics such as material stability and machinability, clamping force, and weight,
- human operations.

These factors result in the machining errors of shape, dimension or quality for a component. For the repeatable machining errors it is easy to measure and model them, and thus to control the errors within the imposed tolerance. For the nonrepeatable errors, however, they are generally neglected mainly because of the difficulties in modelling and control them. The nonrepeatable machining errors take a quite large percentage of the machining errors in precision turning, and which are not negligible for high precision requirements.

Figure 1 shows the proposed approach used for the machining accuracy control system on a CNC turning machine tools. The system is implemented on an IBM PC 486/50. The PC logs in the dimensional data of the component via size gauges during the machining. Based on the logged in data, time series algorithms such as AR and ARMA models are used to model the data and predict the component dimensional variance or machining errors in the further machining. Then the PC send control signals with reference to the predicted errors to adjust driving motors which drive the turning tool to a required position. So better machining accuracy can be achieved. Genetic algorithms are used as an optimisation tool and a classifier to check the unnormal data and forecasting accuracy and optimise the forecasting models. The genetic algorithms are incorporated with the time series models and thus to achieve optimal forecasting and control accuracy, and finally the realisation of intelligent manufacturing.

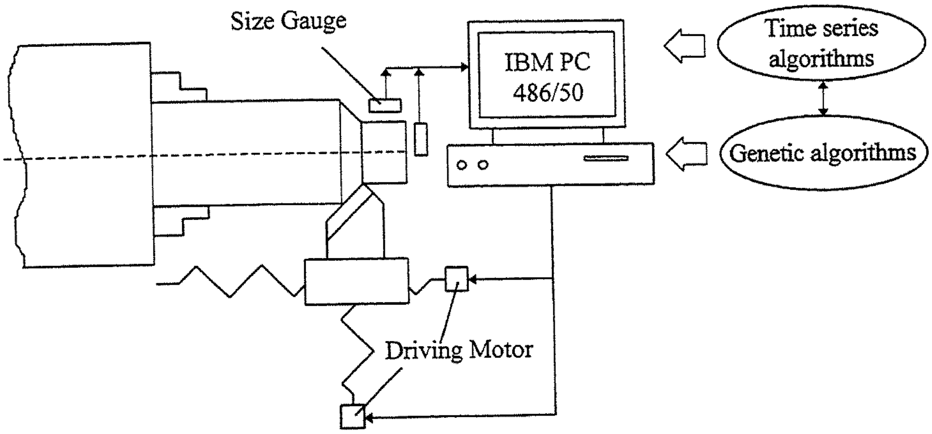


Fig. 1: The architecture of the proposed forecasting and control system

2.2 Time Series Algorithms

In choosing a time series forecasting model, the following factors with respect to turning processes have to considered[2][3]:

- the forecasting form desired,
- the time frame,
- the pattern of data,
- the forecasting speed,
- the accuracy desired,
- the acquisition and availability of data,
- the control actuation speed based on the forecasting models,
- the ease of operation and understanding.

Based on these criteria, AR and ARMA models are chosen as the forecasting and control algorithms. An AR(n) model is expressed as follows:

$$X_t = \phi_1 X_{t-1} + \phi_2 X_{t-2} + \phi_3 X_{t-3} + \ldots + \phi_n X_{t-n} + a_t \tag{1}$$

X_t - dimensional data at time t

ϕ_n - factor

a_t - noise, $a_t \in NID(0, \sigma_a^2)$

Using logged dimensional data $X_{t-1}, X_{t-2}, X_{t-3}, \ldots, X_{t-n}$, the coming data X_t can be predicted with an AR(n) model as equation (1).

An ARMA(n, m) model is expressed as follows:

$$X_t = \phi_1 X_{t-1} + \phi_2 X_{t-2} + \phi_3 X_{t-3} + \ldots + \phi_n X_{t-n} - \theta_1 a_{t-1} - \theta_2 a_{t-2} - \theta_3 a_{t-3} - \ldots - \theta_m a_{t-m} + a_t \qquad (2)$$

X_t - dimensional data at time t

ϕ_n, θ_m - factors

a_t - noise, $a_t \in NID(0, \sigma_a^2)$

Obviously, using logged past dimensional data $X_{t-1}, X_{t-2}, X_{t-3}, \ldots, X_{t-n}$, the coming dimension X_t can be predicted with an ARMA(n, m) model as equation (2). However, the model requires much more computing time which has to be balanced with forecasting accuracy[4][5].

2.3 Genetic Algorithms

The structure of a genetic algorithm is same as that of any evolution program. The genetic algorithms are used to anneal and optimise the time series models during the machining process. For instance, during iteration t, a genetic algorithm maintains a population of potential solutions, $P(t) = \{AR(1)^t, AR(2)^t, \ldots, AR(n)^t\}$. Each solution $AR(i)^t$, i.e. the forecasting and control model, is evaluated to give measure of its 'fitness'. Then, a new population (iteration t+1) is formed by selecting the more fit individuals. Some members of this new population undergo alterations by means of crossover and mutation, to form new solutions, i.e. the optimised solution until getting a satisfactory one. In developing the genetic algorithms, the following factors have to considered at early stage[6]:

- the genetic representation for establishing AR or ARMA models,
- a way to create an initial population of potential models,
- an evaluation function that plays the role of the environment, rating the models in terms of their 'fitness',
- genetic operators that alter the composition of iteration processes,
- values for various factors and parameters that the genetic algorithm uses.

3. SIMULATION OF THE FORECASTING AND CONTROL

Figure 2 shows a screen copy of the simulation on the system illustrated in Figure 1. The simulation can simulate components' dimensional accuracy such as the diameter size and length as defined by an operator. The 'real' machining data are created with random data by a sub-program. With the data, ARMA (8, 7) model is used to model the data and produce the predicted data which can be further used in the forecasting control. As shown in Figure 2, the forecasting accuracy is quite high. An operator can select the type of cutting tool, the material of machining component and model type as he/her requires. The simulation is developed using LabVIEW for Windows which is a user-friendly and highly graphical programming package.

4. FURTHER WORKS

The proposed approach and system are still in the further development. Initial machining trials will be carried out on a CNC turning machine tools. 200 cylindrical components with the dimensions of ϕ 30 mm in diameter and 40 mm in length will be machined. The components dimensional data will be compared with the simulated prediction data. The next goal is to implement the developed algorithms onto the CNC controller and to undertake realistic machining trials with particular reference to the requirements of forecasting and control accuracy of the component geometrical dimensions. To achieve this goal, there are some problems to be overcome such as the executing speed of the algorithms, the driving motors' responding and actuating speeds, and the monitoring of the cutting tool's conditions, and so on.

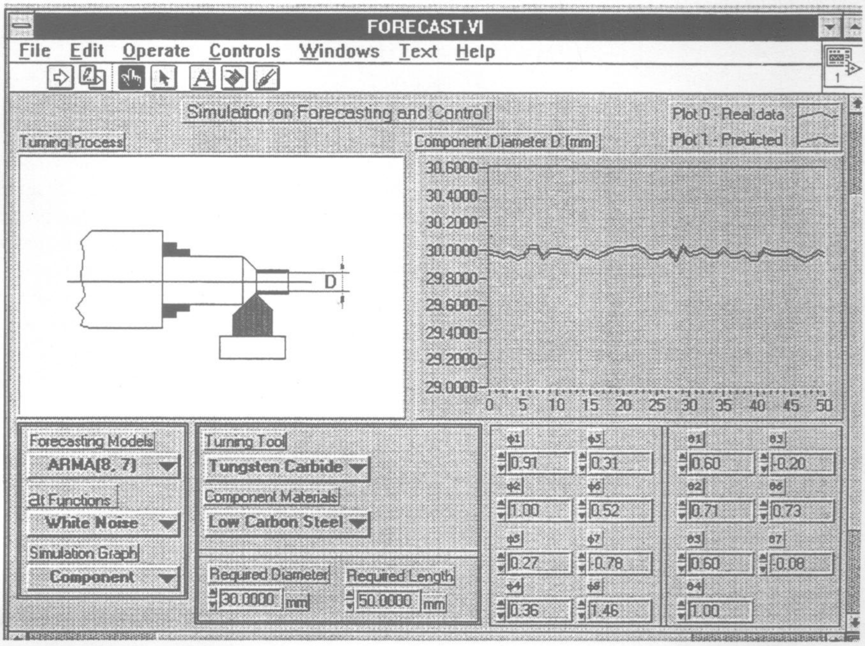


Fig. 2: The screen copy of a simulation on machining accuracy forecasting and control

5. CONCLUDING REMARKS

In this paper the forecasting and control of machining accuracy is explored with the simulation based approach which is economical and efficient. However, the approach and implemented simulations need to be further verified through the machining trials which are being undertaken by the authors. The results will be published in another paper.

6. REFERENCES

1. Groover, M.P., Fundamentals of Modern Manufacturing - Materials, Processes, and Systems, Prentice Hall, 1996.

2. Bowerman, B.L. and O'Connell R.T., Time Series and Forecasting, Duxbury Press, 1979.

3. Pandit, S.M. and Wu S.M., Time Series and System Analysis with Applications, John Wiley and Sons Inc., 1983.

4. C.W. Park, et al, "An in-process flatness error measurement and compensatory control system", Transactions of the ASME: Journal of Engineering for Industry, Vol. 110, pp.263-270, 1988.

5. J.S. Chen, et al, "Real-time compensation for time-variant volumetric errors on a machining center", Transactions of the ASME: Journal of Engineering for Industry, Vol. 115, pp.472-479, 1993.

6. Michalewicz, Z., Genetic Algorithms + Data Structure = Evolution Programs, Springer-Verlag, 1994.

ANALYSES OF RELATIONSHIP AMONG DESIGN METHODS

Ke-Zhang Chen
Department of Mechanical Engineering
The University of Hong Kong
Hong Kong

ABSTRACT

This paper analyzes many existing design methods and their relationship. It is concluded that Axiomatic Design is only one method which illustrates clearly the design principle, design process and design method, and is a design framework. Other design methods belong to a kind of methods for mapping between a special functional requirement (FR) of a product and the design parameters (DPs) of the product in the process of Axiomatic Design. It is obvious that Axiomatic Design method should always be applied to design products. The FRs and DPs should be decomposed into hierarchies from the abstract concepts to the detailed information. During the mapping process, the most effective or suitable "Design for FR" method, according to the specified functional requirements (FRs), should then be used to obtain DPs of the product.

KEYWORDS

Design Theory and Methodology, Axiomatic Design, Design for X, Concurrent Engineering

1. INTRODUCTION

The essence of design is creation. People design some things to satisfy their needs. These things did not exist before and were created by human being. Furthermore, design does not mean only the creation of new products in the form of drawing. Design involves everything, such as new products, processes, software, systems, organizations, even novels and dramas.

Design process consists of two distinct processes: the creative process, where new ideas or solutions are synthesized in the absence of prior examples, and the analytical process, where design decisions must be made by evaluating the new ideas proposed. The creative process depends strongly on the designer's knowledge base and creativity, and is subjective. Therefore, there can be an infinite number of possible creative solutions that can be synthesized to satisfy the same set of requirements. However, the analytical process is determinant and can evaluate the synthesized ideas so as to enable the selection of only good ideas or the best. But both of the processes must be based on a finite set of basic principles, without which the design process is inoperative and has therefore been treated as a mysterious creative process rather than as a rational and systematic activity. Although some among us have produced truly creative breakthroughs, they are the exceptions rather than the rule. Each generation must gain similar experience all over, again and again, and develop its own intuition. Therefore, the field of design needs a science base or principle that can properly guide human endeavor.

In order to obtain the scientific base and to create more things to satisfy more and more requirements from the modem society, much work has been done to understand the design process and develop design theory and methodology (DTM). The DTM has become new area or subject people are much concerned with, although it is an old problem, and many effective design methods have been developed quickly, such as Optimization Design, Reliability Design, Industry Design, Axiomatic Design and so on. But, what is the relationship among them? What should be used for solving a design problem properly? Therefore, it is necessary to analyze the relationship among these design methods to make the most effective use of them.

2. AXIOMATIC DESIGN

According to the theory of Axiomatic Design[1], design involves the continuous processing of information between and within four distinct domains: consumer domain, functional domain, physical domain and process domain(see Fig.1).

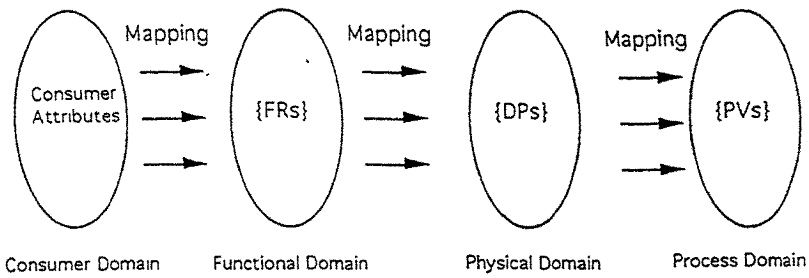


Fig.1 Four domains of the design world

The needs of the customer are established in the consumer domain and then formalized in the functional domain as a set of functional requirements (FRs) that govern the solution process, where each FR is independent of other FRs by definition. The creation of a synthesized solution to satisfy the perceived needs is through the mapping process between the functional requirements (FRs), "What we want to achieve", which exist in the functional domain, and the design parameters (DPs), "How we want to achieve them", which exist in the physical domain. The DPs in the physical domain are then mapped into the process domain in terms of the process variables (PVs). The output of each domain evolves from the abstract concepts to the detailed information in a top-down or hierarchical manner. The hierarchical decomposition in one domain cannot be performed independent of the evolving hierarchies in the other domains. Therefore, the decomposition follows zig-zag mapping between adjacent domains. Fig.2[2] illustrates the zig-zag mapping process between functional domain and physical domain.

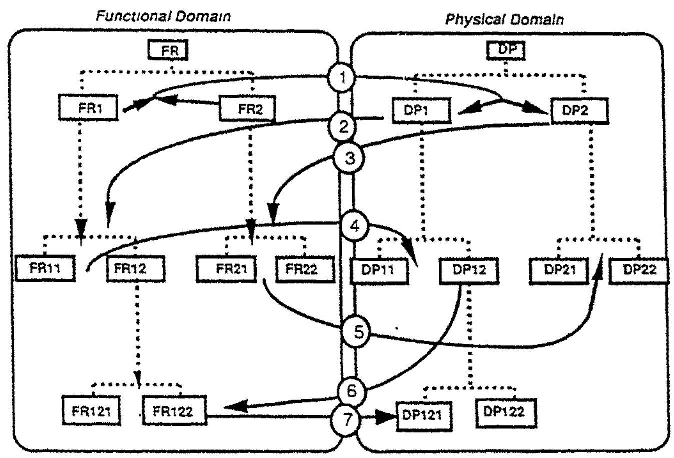


Fig.2 Zig-zag mapping process between functional domain and physical domain

When the design is completed, the product is manufactured by the designed appropriate manufacturing processes and strategies. But, the whole product development process does not necessarily begin from the societal and human needs. It can begin anywhere in the chain. For example, when a new technological breakthrough occurs, it can create a new product as well as new demand, creating markets that had not previously existed. Moreover, the product development process is not planar in two-dimensional space, because the process depends on time and moves forward continuously. For example, when the first round of the design process is completed, and the solution is compared with the original set of FRs and needs, it is highly probable that the additional insight gained through the first iteration may enable the designer to come up with a better set of FRs, which may in turn enable the design and manufacture of more-advanced products. The availability of advanced products may then change the marketing strategy; that is, each time the design loop is crisscrossed, new information is generated which propels all related field to the next plateau. This symbiotic process of the design world, going from "marketing" to "design" to "manufacturing" to "marketing" to "design", etc is a design helix.

In this process of traveling up the "design"-"manufacturing"-"Marketing" helix, a consistent set of decisions must be made. Therefore, there should be some absolute referents that provides the metric for the decisions made at each step. The referents are so called design axioms. The declarative form of the axiom is

Axiom 1 The Independence Axiom
Maintain the independence of FRs.

Axiom 2 The Information Axiom
Minimize the information content of the design.

Axiom 1 states that during the design process, as we go from the FRs in the functional domain with DPs in the physical domain, the mapping must be such that a perturbation in a particular DP must affect only its referent FR. The design process may be expressed as the following design equation:

$$\{FR\} = [A]\{DP\} \quad (1)$$

where {FR} is the functional requirement vector, {DP} is the design parameter vector, and [A] is the design matrix. The design matrix [A] is of the form:

$$[A] = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1n} \\ A_{21} & A_{22} & \cdots & A_{2n} \\ \vdots & \vdots & & \vdots \\ A_{m1} & A_{m2} & \cdots & A_{mn} \end{bmatrix} \quad (2)$$

and each element Aij of the matrix relates a component of the FR vector to a component of the DP vector. In general, the element Aij may be expressed as:

$$A_{ij} = \frac{\partial FR_i}{\partial DP_j} \quad (3)$$

If a design satisfies Axiom 1, its design matrix [A] should be a square matrix, i.e. m = n, and a diagonal matrix, i.e. all the nondiagonal elements are zero. The design is called as an uncoupled design. If the design matrix is triangular, i.e. a square matrix whose non-zero elements occur in a triangular pattern either above or below the main diagonal, which may be represented for three FRs as

$$\begin{Bmatrix} FR_1 \\ FR_2 \\ FR_3 \end{Bmatrix} = \begin{bmatrix} A_{11} & 0 & 0 \\ A_{21} & A_{22} & 0 \\ A_{31} & A_{32} & A_{33} \end{bmatrix} \quad (4)$$

the independence of the FRs can be assured if we adjust the DPs in a particular order; thus Axiom 1 is satisfied. The design is called as a decoupled design. If a design clearly violates Axiom 1, that is, its design matrix is not a diagonal or triangular matrix, this type of design is a coupled design and must be a bad design which should be avoided or re-designed. Therefore, we can choose or obtain the acceptable design by using Axiom 1.

Axiom 2 states that, among all the design that satisfy the Independence Axiom (Axiom 1), the one with minimum information content is the best design. Information content (I) was defined in terms of the probability (P) of attaining a set of FRs as:

$$I = \log_2 (1/P) \quad (5)$$

The useful information is that related to achieving a given FRs. When there are a number of discrete events occurring in an ensemble with probabilities P_i, the average information content of the discrete events is given by:

$$I = -\sum_i P_i \log P_i \quad (6)$$

In a design situation, information can be broadly interpreted to be

$$I = \log_2\left(\frac{System\ range}{Common\ range}\right) \quad (7)$$

where system range is the capability of the proposed solution and common range is the range of overlap between system range and the design range which the designer is trying to satisfy. The information is nondimensional and, therefore, the minimum information criterion can select the best solution among those proposed, regardless of the number of variables with different units involved.

Therefore, Axiomatic design illustrates clearly the design principle, design process and design methods. This method guides the creative process with mapping and decompositon, and the analytical process with two axioms. Particularly, this method requires a zig-zag mapping between adjacent domains which implies a concurrent engineering by people with different disciplines. This method is a scientific foundation for design field so as to provide a fundamental basis for the creation of products, processes, systems, software, organizations and so on. This is a significant departure from the conventional design process, which has been dominated by empiricism and intuition.

3. OPTIMIZATION DESIGN

Optimization Design[3] is an efficient method which can be used to find out the optimum values of the objective function, f(X), and every variables, $x_1, x_2, \ldots, x_n$, affecting the objective function for a given structure or conception. The first step of the method is building a mathematical model of the design problem. The typical mathematical model may be expressed as:

$$\begin{aligned} &\text{Minimize } f(X), \quad X = [x_1, x_2, \ldots, x_n]^T \\ &\text{subject to the constraints:} \\ &\quad g_i\ (X) \le 0, \qquad i = 1, 2, \ldots, m \\ &\quad h_j\ (X) = 0, \qquad j = 1, 2, \ldots, p \end{aligned} \quad (8)$$

The second step of the method is the iterative numerical optimization calculation which can obtain the optimum values of the objective function and the related variables by using a suitable optimization calculation method.

Actually, the objective function is a functional requirement, that is, FR mentioned in Axiomatic Design above, and those variables affecting the objective function are the design parameters, that is, DPs. But, in this case, FR must be able to be expressed by DPs in form of mathematical equations. Therefore, this method is just a method mapping between one FR and DPs at lower or lowest level of FR and DP hierarchies during the process of Axiomatic Design.

For the design with only one FR, the optimum design does not involve the independent requirement of the FRs and does not violate Independence Axiom. Therefore, this method is consistent with the Axiomatic Design and thus is very efficient. But, for the design with multiple objective functions, the optimization design is not very efficient. In this case, it, usually, selects the most important FR as an objective function and eliminates the others, or gives the different weights to the different objective functions and forms one composite objective function. For the first method, it becomes a one-FR problem and the optimization design is efficient for finding out the corresponding DPs as previously explained. For the second method, the multiple objective functions mean more than one FR and are controlled by the same set of DPs. Therefore, the FRs must not be independent with each other. The design matrix, according to Axiomatic Design, must not be a diagonal matrix and thus the design must not be an uncoupled design. Although it is possible to be a decoupled design, the optimization results are rather unconvincing by just giving the different weights to different objective functions. In this case, the new FRs must be re-selected to satisfy Axiom 1, the optimum design can be used to find the optimized DPs for each FR, and Axiom 2 should then be used to get the optimum solution for all FRs, because Axiom 2 can efficiently select the best solution regardless of the number of variables with different units involved.

4. ROBUST DESIGN

Robust Design[4] divides the design into three stages: conceptual design, parameter design and tolerance design, and illustrates mainly the last two stages. This method focuses on the two functional requirements: the lowest cost and the most stable quality. Since this method usually selects the material with the lowest price and/or the lowest class of elements first and then find out the values of the control factors(i.e., DPs), it's design equation, according to Axiomatic Design, can be expressed as:

$$\begin{Bmatrix} \text{The lowest cost} \\ \text{The most stable quality} \end{Bmatrix} = \begin{bmatrix} X & 0 \\ X & X \end{bmatrix} \begin{Bmatrix} \text{Selection of materials or class of the parts} \\ \text{Selection of the values of the control factors} \end{Bmatrix} \quad (9)$$

It is obvious that this is a decoupled design and that this method satisfies the Independence Axiom of Axiomatic Design.

The parameter design of this method introduces Noise Factors, that is, those variables in a design that are uncontrollable or unpredictable and represent an uncertainty that the desired output response will be achieved. The output response may be represented by a function·

$$y = g(M, Z, R) + e(X, M, Z, R) \quad (10)$$

where g is predictable, e is unpredictable, M is input or signal factor, Z is control factors, R is scaling factors which can be altered to achieve a desired relationship between a signal and output response, and X is the noise factors. To ensure the stability of quality, the Z and R can be chosen such that the predictability ratio $\Delta g/\Delta e$ is maximized. The signal-noise ratio (SN ratio) is the $\log_{10}$ of this predictability ratio, multiplied by 20, and should be maximized too. SN ratio can be determined by statistical technique. Orthogonal arrays are used in design of experiments to assure a representative sample of different combinations of the levels of the control factors. The SN ratio as a function of the output response samples $y_1, y_2, ..., y_n$ can be expressed as:

$$\eta = 10 \log_{10} [(S_m - V)/ nV] \quad (11)$$

where $S_m = (\Sigma y)^2/n$. V is the total variation of y over the entire range of n samples and n is the number of samples. The maximization of SN ratio can be used to determine the best combination of the values of the control factors(i.e., the best set of DPs). Therefore, the method is also a kind of optimization design, but objective function or FR is the maximized SN ratio which is to reduce the effect of the unpredictable component on the output response to obtain stable quality. Therefore, this method is also only a method mapping between a FR(stable quality) and DPs in the process of Axiomatic Design.

Tolerance design is similar to parameter design. The differences between them are that the different combination of experiments is for the different levels of tolerance of the most notable control factor, not for the different levels of the control factors, and that its objective function is the output response rather than the SN ratio. The goal of the experiments is finding out the most notable tolerance factor. Finally it makes compromise between the cost increased by the reduced tolerance and the benefits increased by the raised quality to obtain the best tolerance of the most notable factor.

To sum up, Robust Design is also a efficient method mapping between a FR and DPs at the lower or lowest level of FR and DP hierarchies during the process of Axiomatic Design and is consistent with Axiom 1 of the Axiomatic Design. Its functional requirements focus on the lowest cost and the most stable quality. By using this method, the best combination of the values of control factors (DPs) and the best tolerances of the notable factors can be determined.

5. RELIABILITY DESIGN

In order to ensure that the products or components can be used reliably, the strength of the component must be greater than the stress acting on the component with respect to a certain mode of failure. In fact, for the strength, the properties of the material used and factors affecting the strength, such as dimensional tolerances, the surface finish and surface treatment, are all variable within a certain range. For the stress, the load and the factors affecting the stress, such as stress concentration and temperature are also variable within a certain range. Therefore, the strength of the component and the stress acting on the component are all variable and have their probability distributions. Thus the strength and the stress may have interference as shown in Fig.3[5]. In the interference area, the strength is less than the stress and the component will be damaged or failed. Therefore, although the mean value of the strength, $\bar{\delta}$, is greater than the mean value of the stress, $\bar{s}$, there is the possibility of failure. If the probability of failure is big, the reliability is poor. Reliability Design has been developed to reduce the possibility of failure or to reach the required reliability. This method is based on probabilistic data, such as the strength degradation data for material to be used and the design data

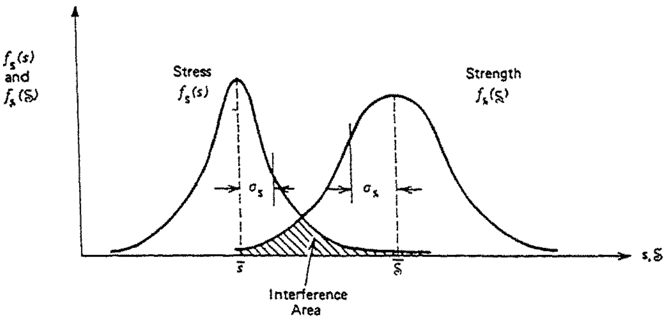


Fig.3 Stress-strength interference

on the statistical distribution of loads. For the normally distributed strength and stress, the method uses a coupling equation developed from the lower limit of the standard normal variate, as follows:

$$Z = -\frac{\bar{\delta} - \bar{s}}{\sqrt{\sigma_\delta^2 + \sigma_s^2}} \tag{12}$$

where Z is the standard normal random variable, $\bar{\delta}$ is the mean value of the strength, $\bar{s}$ is the mean value of the stress, and σ_δ and σ_s are the standard deviations of strength and stress respectively. According to the σ_δ, σ_s, $\bar{\delta}$ and $\bar{s}$, the value of Z can be calculated and, hence, the reliability of the component is found from the normal tables. If a certain value of reliability for the component is required, the coupling equation and normal tables can be used to design the component according to the value of reliability required.

Therefore, this method is special for the reliability of products which is one of the functional requirements(FRs) and does not represent all FRs. Reliability Design is thus also only a design method mapping between a special FR (reliability) and DPs during the process of Axiomatic Design.

6. DESIGN FOR X

Among design methods, there are many design methods the names of which are in terms of "Design for X", such as Design for Manufacturability, Design for Assembly, Design for Maintainability, Design for Serviceability, Design for Logistics and so on. X is a general representation for manufacturability, assembly, maintainability, serviceability, logistics and so forth. In fact, X is the functional requirement, i.e. FR mentioned in Axiomatic Design above. It is obvious that all these design methods are the special methods mapping between a special FR (i.e. X) and some DPs during the process of Axiomatic Design. Therefore, Design for X should be called Design for FR. Table 1 lists some of these design methods and their functional requirements. With increment of requirements from the modern society, this kind of design methods have been developing continuously. Most of these methods offer some guidelines which are, actually, sub-functional requirements. Therefore, these methods need to be developed further and can be developed by using Axiomatic Design. Some of these methods also cover other methods, such as Design for Availability. The term availability combines the concepts of reliability and serviceability, and recognizes the fact that perfect reliability is not feasible. Thus Design for Availability includes Design for Reliability and Design for Serviceability. Since one functional requirement can be decomposed into several sub-functional requirements which can be decomposed further, it is normal to have this overlap of different design methods. The X may be at any level of FR hierarchy. But these methods can get the solution satisfying only its X at certain level of FR hierarchy, not all FRs. Therefore, in order to get the solution for all FRs, we have to use Axiomatic Design method and, during the zig-zag mapping process, to choose the most suitable or effective "Design for FR" method to obtain DPs of the product according to the specified functional requirements(FRs).

Among design methods, there are also other design methods the names of which are not in terms of "Design for X", such as Industry Design, Green Design, Value Engineering, Fuzzy Design and so on. But, after analyzing, it can be known that these design methods are also the special methods

Table 1 Some of Design for X and their FRs

Name of the method	FRs
Design for Manufacturability	Easy and economical manufacture
Design for Assembly	Easy and economical assembly
Design for Disassembly	Easy and economical disassembly
Design for Maitainability	Maintenance with the minimized cost, inconvenience & effort
Design for Serviceability	No or little service which is easy and economical
Design for Availability	Continued performance without excessive down-time
Design for Logistics	Logistics- friendly
Design for Security	Protection against danger and unauthorized access
Design for Vandalism Prevention	Preventing the willful damage directed at public property
Design for Safety	Without injury to people and damage to equipments
Design for Usability	User-friendly
Design for Low-quantity Production	With lowest cost of tooling, production equipment & material
Design for the Physically Disadvantaged	Convenience for the disabled and the elderly
Design for Testability	Easy to isolate faults, and to write and execute test program
Design for Short Time-to-Market	With shortest interval between the decision to develop it and the point when it is available for sale in the market

mapping between a special FR and some DPs. Table 2 shows the FRs these methods map from and their rewritten names into the form of "Design for X".

Table 2 The FRs of some design methods and their rewritten names

Name of the Method	FRs	Re-written names
Industry Design	Having good appearance	Design for Appearance
Green Design	Facilitating environment protection	Design for Environment
Value Engineering	Having the maximized value	Design for Value
Fuzzy Design	Fuzzy FRs	Design for Fuzzy FRs
Robust Design	Having high Robustness	Design for Robustness
Reliability Design	Having high reliability	Design for Reliability

7. CONCLUSION

Although there appear many design methods, Axiomatic Design is only one method which illustrates clearly the design principle, design process and design method, and, therefore, it is a design framework. Other design methods belong to a kind of methods which is for mapping between a special FR and some DPs in the process of Axiomatic Design, and can be named as Design for FR. Therefore, the relationship among design methods can be graphically represented in Fig.4.

It is obvious that Axiomatic Design method should always be applied to design products step by step. FRs and DPs should be decomposed into hierarchies from the abstract concepts to the detailed information. At the beginning of the mapping process at top level of hierarchies, it is very important that Creative Design method should be used to get the good concept of the product. Then, during the zig-zag mapping process, the most effective or suitable "Design for FR" method, according to the specified functional requirements (FRs) of the product, should then be used to obtain some DPs of the product. When the design problem can be written as a mathematical model at the lower or lowest level of hierarchies, Optimization Design is helpful to get optimum DPs for any FR. Of course, the whole design process also can be operated by using computer in the future. Thus, the content of Computer-Aided Design (CAD) is much more than that people think of now. Before that, there is a lot of research work to do, including the integration of the existing software of certain design methods and the programming of software of new design methods, such as Computer-aided Axiomatic Design and Computer-aided Creative Design.

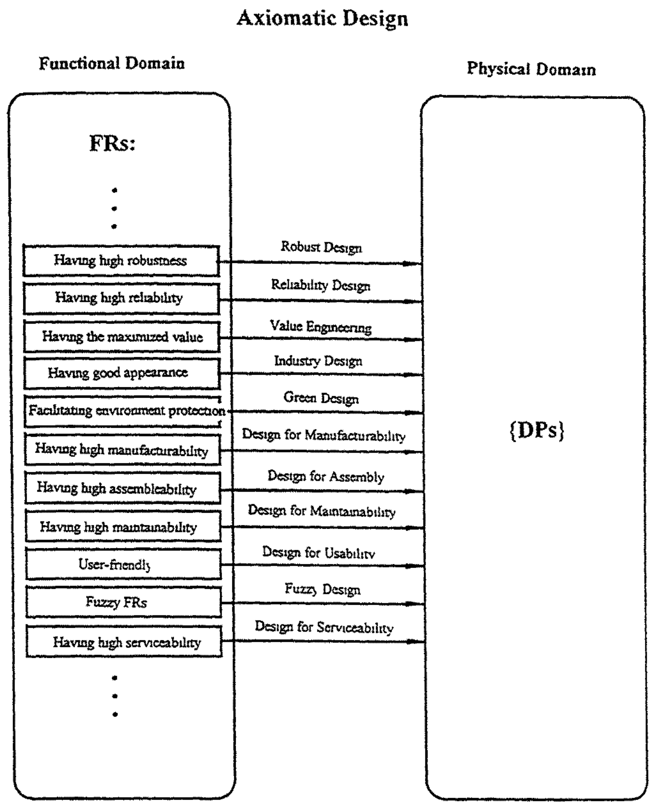


Fig.4 Relationship among Axiomatic Design and other design methods

ACKNOWLEDGMENTS:

I would like to take this opportunity to express my sincere acknowledgment to Professor Nam P. Suh, Head of Department of Mechanical Engineering in Massachusetts Institute of Technology, U.S.A , for his advice and assistance during the period of this work when I was visiting MIT.

REFERENCES

1 Suh, Nam P., The Principles of Design, Oxford University Press, New York, 1990.
2 Kim, Sun-Jae, Suh, Nam P., and Kim, Sang-Gook, "Design of software systems based on Axiomatic design", Robotics & Computer-Integrated Manufacturing, Vol.8, No.4, pp.243-255, 1991.
3. Rao, Singiresu S., Engineering Optimization: Theory and Practice, Third Edition, John Wiley & Sons, Inc., 1996.
4. Phadke, Madhav S., Quality Engineering Using Robust Design, Englewood Cliffs, N. J.: Prentice Hall, 1989.
5. Kapur, K. C. and Lamberson, L.R., Reliability in Engineering Design, John Wiley & Sons, 1988.
6. Helander, M and Nagamachi, M., Design for Manufacturability, Taylor & Francis Ltd., 1992.
7. Boothroyd, G., Assembly Automation and Product Design, Marcel Dekker, Inc., 1992.
8. Mather, H., "Logistics in Manufacturing: A Way to Beat the Competition", Assembly Automation, Vol. 7, No. 4, pp. 175-178, 1987.
9. Bralla, James G., Design for Excellence, McGraw-Hill, Inc., 1996
10. Lindbeck, John R., Product Design and Manufacture, Prentice-Hall, Inc., 1995

FEATURES AS AUTONOMOUS AGENTS: AN ALTERNATIVE PARADIGM FOR CONCURRENT ENGINEERING.

D. Jacquel, J.C. Salmon and F.G. Mill
Department of Mechanical Engineering
The University of Edinburgh, EH9 3JL, SCOTLAND

ABSTRACT

This paper describes a novel approach to the design of concurrent engineering systems by reversing the traditional view of such a system as a number of distinct but integrated modules operating on a data structure that is the product model. In this traditional view the data structure is a passive entity, and must be acted upon by modules such as design, process planning and NC generation.

In this new approach we imbue the model with intelligence (or at least a degree of autonomy), i.e., have an active model surrounded by passive expert modules which are capable of answering questions appropriate to their area of expertise. Indeed the model is allowed to be composed of any number of active agents, each responsible for an appropriate portion or feature of the model. Therefor ea multi-agent system is created in which each design feature is an agent that tries to successfully design, process plan and generate NC code for itself.

KEYWORDS

Feature Oriented Engineering, Agents, Multi-agent System, Active Model, Concurrent Engineering

1. INTRODUCTION

This paper focuses on a new approach for the design of a concurrent engineering system. This new approach takes its roots in the Artificial Intelligence field, and consists of applying the emerging multi-agent paradigm to the design of a feature based design system.

In the traditional approach, a concurrent feature based manufacturing system is an integrated set of software modules, such as modeller, geometric reasoner, process planner and NC code generator. They are all acting upon a passive data structure that is the product model, continuously enriching it so a design can evolve from the conceptual stage to a finished product complete with methods of production. Using this classical approach, the designer creates a product through a Design/ Evaluation/ Redesign loop, adding new features to the model before using one of several modules to evaluate the quality of the design against chosen criteria. This evaluation process can pinpoint flaws in the design, which can then be corrected before enriching the model with new features.

No matter how complex and powerful the modules used on the model are, the classical design process is not an interactive, concurrent activity but merely a client/server exchange between system and designer. The system will give answers to specific requests but is incapable of taking initiatives in the design process.

It is proposed to reverse this traditional approach, and create a system with an active product model and passive modules. This is achieved by applying the increasingly popular multi-agent paradigm to the system architecture. Indeed each instance of a design feature is an autonomous agent in the system. The product model therefore becomes an active community of feature agents that communicate and cooperate with one another in order to accomplish their goal: *engineer themselves*. Thus, the design module looks like a traditional design system from the user's point of view, but unbeknownst to the user, the roles have been reversed. The user firmly believes that they are in control and are designing some component for their own purpose. The product agent on the other hand is 'using' the designer as an expert tool to fill in details about the design. In this way, user and product work in a symbiotic relationship to achieve a common aim.

An important factor is the interrelationship between the feature agents making up a complete product design. Certain features may intersect with each other or interact with each other in interesting ways that place considerable constraints on the manufacturing solutions for individual features. This problem is solved using two mechanisms. Firstly, all agents must be able to cooperate within the product model that they inhabit. The analogy here is one of a community and though each agent wishes for success as an individual, success of the entire community is paramount. Co-evolution techniques leading to emergent near-optimal solutions is an active research topic in Artificial Intelligence the results of which will be drawn on for applications in the concurrent engineering domain [1].

2. AGENTS

Formalising an agent definition is a difficult task, despite, or indeed because of, the vast amount of activity currently in the field. It is important distinguish the nature of agents at both the agent level and the community level.

2.1 Description of an Agent

Many recent publications related to agents and multi-agent systems (MAS) propose to give different definitions for agents [2][3][4]. At the entity level the most important features of a software agent are its co-operation and sociability aspects. An agent, can be thought of as an entity working inside a community and co-operating with other agents to achieve a goal. Rather than defining the nature of an agent, agency is indirectly described through properties required of agents.

- **Autonomy**: An agent can act with a certain range of autonomy and is capable of spontaneous actions. An agent has the capacity to plan its own actions and follow a self-prescribed schedule. Such autonomy can only be achieved through both synchronous and asynchronous actions.

- **Communication**: An agent must be able to carry bi-directional conversation with other agents in a language rich enough to allow it to express intentions and abilities to the community. Moreover, in order to comply with the autonomy property the communication protocol used by an agent should break the client/server protocol and permit peer-to-peer dialogues. [5]

- **Co-operation**: Using their communication abilities, agents should be able to initiate dialogues with one another in order to achieve their goals. This co-operative behaviour should lead to improved reactivity of the system and additionally, better interaction with users.

2.2 A Multi-Agent System (MAS)

Only considering the notion of agency at the agent level ignores the social dimension of agents. A social software agent is only useful if living inside a community of agents. In such a community, the asynchronous nature of each agent leads to exchanges of messages and decision/actions throughout the system. This asynchronous, peer-to-peer activity between individuals inside the system results in an overall multi-goal, co-operative search for an optimum solution.

Unlike a traditional procedural synchronous system, each feature agent initiates a dialogue only on an asynchronous basis, when it needs to solve a problem or improve its fitness against its goals. This allows the system to only use its processing power on critical parts of the design model, resulting in improved resource allocation.

2.3 Agents and Concurrent Engineering

Concurrent engineering (CE) proposes to make product creation a faster and more efficient process by allowing traditional sequential tasks such as geometry design and process planning to take place simultaneously. By parallelising the different tasks of the engineering process, CE brings early feed back to each task, avoiding costly and time-consuming redesign phases. The multi-agent paradigm and techniques have already been applied in a number of ways to CE [6].

Agents can be used as a mean of integration between the existing engineering tools by using autonomous interface agents between design and process and manufacturing applications [7]. In this approach agents provide their communication and information sharing capabilities to create a dialogue between design and manufacturing, bringing early feedback about the manufacturability of a product being designed.

Using more than interface agents [8] proposes integration of design, manufacturability analysis, incremental process planning, dynamic routing, and scheduling, using both feature agents and module agents (geometric interface, design agent, part agent and machine agents).

This work has arisen out of a previous project entitled *Simultaneous Engineering System for Applications in Mechanical Engineering* (SESAME), (BRITE/EURAM 0565). The goal in SESAME was a Simultaneous Engineering Workstation (SEW) and attempted to unify the tasks of Computer Aided Design, Process Planning and NC Generation in such a way that they could be used by a single engineer on a single seat in a standard environment. This goal SESAME achieved with a significant degree of success [9], but the tasks were still performed in a sequential manner as is evident from the architecture in Figure 1. The system was not truly concurrent, but achieved greater integration than earlier process planning systems.

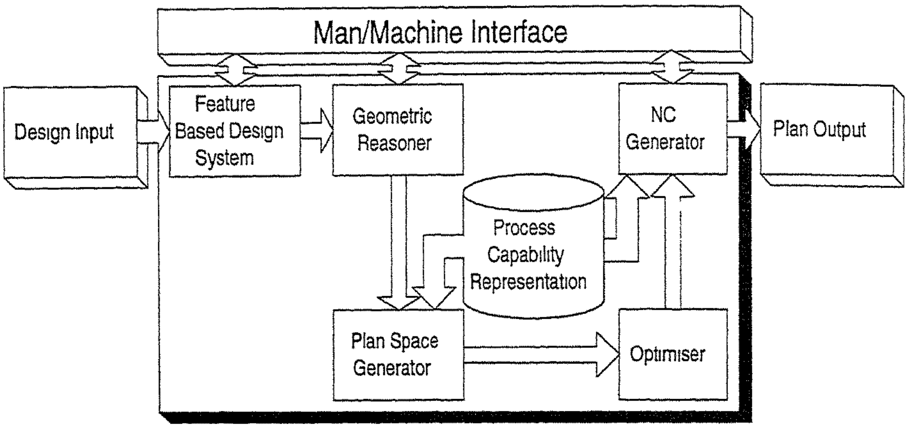


Figure 1: SESAME Architecture

A subsequent proposal in order to increase the concurrency of the system was to build intelligent agents capable of incremental design, process planning and NC generation. This coarse grained approach (Figure 2) can be seen in a number of systems [10], and is a valuable approach when trying to integrate existing products into a concurrent agent based systems.

The Edinburgh 'Features as Agents' approach (Figure 3) uses a fine-grained model with lightweight agents acting for the features in the design and being aided by expert assistant modules such as incremental design, process planning and NC generation systems. This method is not suited to adapting existing commercial systems, but the knowledge gained in the group from SESAME and other projects is present to enable existing modules to be rewritten and adapted.

This fine-grained approach allows solutions to the current design problem to be developed all the time in the dead time between the human's design decisions. The fact that many of these solutions may be unsatisfactory in the finished design is unimportant as the system always holds as complete a picture as it can at any point, and after the last design decision is made, little work need be done to take the current plan and adapt it to the final design (in the majority of cases). In addition the designer is kept constantly aware of the downstream implications of their design.

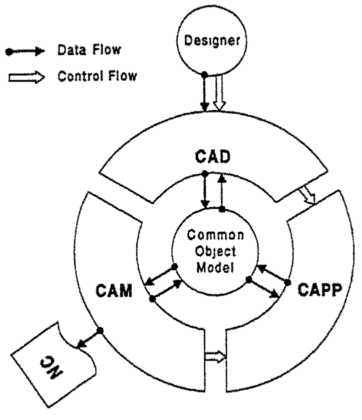


Figure 2: Coarse Grained Agent Model

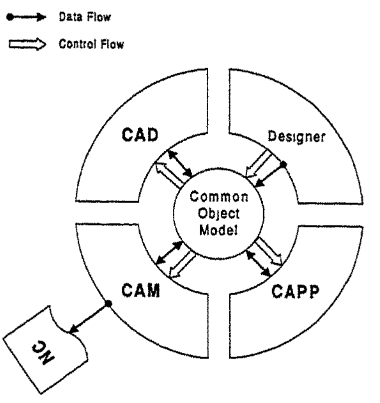


Figure 3: The 'Features as Agents' Model

3. THE EDINBURGH 'FEATURES AS AGENTS' PROPOSAL

The SESAME project was a major improvement over traditional systems, offering a highly integrated set of tools under a single interface allowing a user to cover the complete design process with rapid feed back about manufacturability and permitted the generation of NC code. However the design process, even though integrated, remained sequential rather than concurrent.

The Edinburgh proposal advances the benefits of SESAME by replacing the static product model with an active representation. This can be achieved with the multiagent paradigm applied at low 'fine-grained' level of design features. This radical switch underneath the interface is bringing new benefits to the system. It is also an efficient way of applying and solving design constraints as they can be made part of each agent's goal. It should also ensure quality models at all time and allow complex questions to be answered quickly as agents are constantly working on behalf of the user to find an optimum solution. This constant activity generates much redundant information, but this information is generated in otherwise idle CPU time, increasing the system's apparent performance.

3.1 Small example

To illustrate the proposal, a simple example of how the system works during the design of a small prismatic component is now presented.

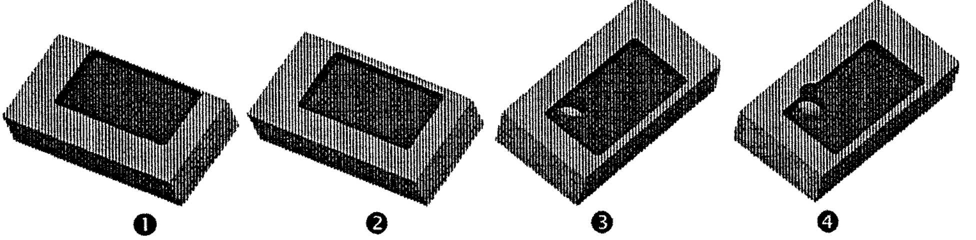


Figure 4: An example component design sequence

Table 1 below shows the evolution of the system during the design process pictured in Figure 4 at three different levels. The user level reflects the designer interaction with the system. The model level represents the product state in terms of geometry. Finally the agent level focuses on the state of and communication between the agents populating the model.

This small example illustrates two important concepts not found in traditional CAD systems. It displays automatic resolution of a thin wall problem within the constraints defined by the user (*). It also illustrates the increased interactivity by allowing an agent to initiate dialogue with the user in the case of an unresolved problem (**).

Table 1 Agent Communication during Design Example

User level	Model Level	Agent level (① shows inter-agents communication)
Creation of blank block with tight size and position constraints	Untouched blank	**block**: I'm the only agent in the model ... sleep
Add new pocket with normal position constraints	❶	**pocket**: I'm not alone in the model. **Pocket①***Broadcast*: New pocket at position xyz **block**: thin wall problem, but I can't move/resize **block①***pocket*: you're creating a thin wall
	❷	**pocket**: I'm moving to new position xyz (*) **pocket①***Broadcast*: pocket at new position xyz **pocket**: no more problem ... sleep **block**: no more problem ... sleep
Add new hole with tight constraints on position	❸	**hole**: I'm not alone in the model. **Hole①***Broadcast*: New hole at position xyz **block**: I can't guarantee access, I can't move/resize **block①***hole*: access problem block position xyz **pocket**: the new hole interacts with me. **pocket①***hole*: there is a pocket at position xyz **hole**: try to solve access problem, but I can't move/resize **hole①***pocket*: Can you move to new position xyz? **pocket①***hole*: I can't comply with request. **hole**: I can't solve this problem. **hole①***user*, I have an access problem. (**)
Receive message from hole. Modify hole position	❹	**hole①***Broadcast*: hole new position xyz **block**: no more problem ... sleep **pocket**: no more problem ... sleep

3.2 Current state of work

Using the Feature Based Design knowledge gained in SESAME, a new design tool is being implemented as a multi-agent system. It takes a fairly simple subset of the problems to be solved during the conception process in order to test the validity of our approach.

To an external user, the new implementation appears little different from the previous Feature Based CAD system. It allows the user to add negative (material removal) features to a blank to obtain the desired design. Underneath the user interface, however, radical changes have taken place. Where there use to be a passive product model on which software modules were acting, there is now a community of agents tirelessly working together to reach an optimum solution.

Each time the user adds a new feature to his design, they actually gives birth to a new software agent that join the existing community inside the system. As soon as this new feature/agent enters the MAS, it can start following its own plans and interact with other agents in order to satisfy its goals. It is this underlying community of agents that represent the living product model.

The current MAS is limited to geometric reasoning such as thin walls detection but it already shows its potential by immediately detecting problems during design. For example, if the user places a new hole too close from another one, creating a thin wall, the system will immediately detect it and take action to solve the problem. This could lead to one of the agents deciding simply to move slightly aside or reduce its diameter (subject to previously supplied constraints), or ask the user for an alternate solution. The system also delivers better performance in terms of response time when adding new features to an existing design because it is incremental by nature. It doesn't need to re-analyse the entire design against the new feature, the new agent interacts locally with existing agents only. Of course a snowball effect is always possible when adding a new feature to the design but this only happens if agents are insufficiently constrained, or if the new feature interacts with many others.

3.3 Future work

The first addition to the current implementation will be to add non-feature agents for conflict resolution and solution proposal. These new agents should share the same structure as the feature agent but provide more complex computation power to propose solutions to problems our light-weight feature agents are unable to solve. A second addition will be expert modules to handle support for process-planning and NC code generation to our current agents. This should not be a major difficulty thanks to the modularity of the Edinburgh approach. Adding support for these tasks also requires adding new knowledge/abilities to our existing agents.

It is already possible to express constraints local to a single agent, the next step is to work on constraints propagation inside the agent community forming the product model. It is also intended to investigate the feasibility of applying the multi-agent paradigm to a constraint solving feature based design system that would automatically generate design solutions. Lastly, we will investigate solutions to the problem of combinatorial explosion that we can see looming in the system.

4. CONCLUSION

This new approach, consisting of turning the system upside-down, making each feature an active element of the system has, so far, shown potential. The constant activity of the agent community radically changes the way the system behaves. The active model, always in search for an optimum solution, ensures a quality model at all stage of design. Leading to several major improvement from the traditional design system architecture.

One could argue that constantly analysing the model generates tremendous load on the computing resources but the MAS provides a better use of computing resources compare to the traditional approach. Indeed, it is believed that, despite the large increase in term of number of operations, the active model provide better overall performances by diluting the calculations along the entire design process.

The Design-Evaluation-Redesign loop no longer needs to be performed by the user. This process is achieved individually by each agent in the system during the design phase. Being able to detect potential problems at early stages through this constant self-assessment of the model, the multi-agent approach reduces costly redesign. The active model is also an efficient way of applying constraints to the design since the constraints can be embedded inside each agent. Finally a higher interactivity during the design process is achieved by enabling any agent to initiate a dialogue with the user.

5. REFERENCES

[1] Husbands P., Mill F., 'Simulated Co-evolution as the Mechanism for emergent planning and scheduling', Proceedings of the Fourth International Conference on Genetic Algorithms, Pages 264-270, Morgan Kaufmann Publishers, San Diego CA, July 13-16, 1991.

[2] Wooldridge M, Jennings NR, 'Intelligent Agents: Theory and Practice' , Knowledge Engineering Review, 10(2):115-152, 1995.

[3] Genesereth MR, Ketchpel SP, 'Software Agents', Communications of the ACM 37(7): 48-53, July 1994.

[4] Franklin S, Graesser A, 'Is it an Agent, or just a Program?: A Taxonomy for Autonomous Agents', Proceedings of the Third International Workshop on Agent Theories, Architectures, and Languages, Springer-Verlag, 1996.

[5] Petrie CJ, 'Agent-based Engineering, the Web, and Intelligence', IEEE Expert. 1996.

[6] Brown DC, Lander SE, Petrie CJ, 'The Application of Multiagent Systems to Concurrent Engineering', Concurrent Engineering: Research and Applications, 4(1): 2-5, March 1996, Technomic Publishing Co Inc, 1063-293X

[7] Frost RH, Cutkosky MR, 'Design for manufacturability via agent interaction', ASME Design for manufacturing Conference, paper number No 96-DETC/DFM-1302, Irvine CA, August 18-22, 1996.

[8] Balasubramanian S and Norrie DH, 'A Multiagent Architecture for Concurrent Design, Process Planning, Routing, and Scheduling', Concurrent Engineering: Research and Applications, 4(1): 7-16, March 1996, Technomic Publishing Co, 1063-293X

[9] Mill F. G., Naish J. C., Salmon J.C., 'Design for machining with a simultaneous-engineering workstation', Computer-Aided Design, 26(7): 521-527, July 1994.

[10] Wunderli M, Norrie M; Schaad W, 'Multidatabase agents for CIM systems', Int. J. Computer Integrated Manufacturing, 9(4): 293-298, 1996, Taylor & Francis, 0951-192X

KEY ENABLING TECHNOLOGY FOR CONCURRENT ENGINEERING

Brian C. Kuttner
C-TAD Systems, Inc.
Ann Arbor, MI USA

ABSTRACT

COMMUNICATIONS IN THE AGILE ENTERPRISE

What is the "glue" that holds the Agile Enterprise together? What prevents "Just in Time" from being late? Obviously the answer is communication. But what sort of communication? It is a tautology to say "Agile Communication" but in fact that is exactly what is called for. There are all sorts of different forms of communication, ranging from the verbose to the obtuse. The term "Agile Communication" denotes communication that absorbs the least communication time between the greatest number of agents. CAD Conferencing appears to offer this capability.

1. INTRODUCTION

In their paper entitled "Communication, Processing and Idleness in Concurrency - an Investigation of Agile Manufacturing", T C. Woo *et al* [W96] have provided a convincing framework for the analysis of manufacturing agility based on the theory of communication developed by Shannon [S48] and others

Where Shannon wrote that.

$$\text{Information} = - \Sigma\ i \log p_i \ (p_i = \text{probability of event}) \qquad (1)$$

Woo argues that:

$$\text{Communication between A manufacturing agents} = A \log A. \qquad (2)$$

Furthermore, Woo found that approximately 1/3 of the time involved in manufacturing a product is spent in communication - or coordination.

This paper disagrees only with the premise and conclusion of Woo *et al.*

2. PREMISE

Woo *et al* believe that the objective of teaming is to find the mechanics by which

$$1 + 1 > 2$$

In the present paper the objective is somewhat less ambitious. Simple equality would suffice but realistically, there will always be some "noise" in any large scale enterprise and the objective therefore is to minimize the noise.

3. COMMUNICATION IN A CONCURRENT ENVIRONMENT

The analogy between the engineering/manufacturing process to the classical bucket brigade problem is obvious. Consider the classic example of the bucket brigade. Let's say 10 men are available to extinguish a fire 100 meters from a water source. The objective is obvious: to put out the fire in the least amount of time. One solution is for the ten men to all rush at the spigot simultaneously and then dash for the flames. This solution has the merit of low overhead and enthusiastic worker involvement. It also allows for the possibility of one or two exceptional athletes in the group to achieve above average speed. However it lacks *coordination*. The time they will spend bumping into each other at the spigot far exceeds the potential saving of the athletes in the crowd.

The near optimum solution can be shown mathematically to be accomplished by stationing the men 10 meters apart and handing off the water a bucket at a time while simultaneously returning the empty buckets along the same chain As a refinement, the superior athletes could be spaced further apart and the slower ones closer together. If the volume of water allows, more than one chain may be set up etc. Note that the total amount of work accomplished is exactly the same - whether we use the refined bucket brigade or the "dash for the flames" approach (Work = Force x Distance). The only difference is the speed We call this relative optimum "agility", and the communications that make it possible "*Synchrony*".

4. TIME TO MARKET

By analogy, a given product can be manufactured in a multitude of ways, each with its own plusses and minuses. The optimum from the point of view of Agility, will be the one that can be replicated to achieve the shortest time to market, at the least cost This method can be shown to be the one with the highest degree of *Synchrony*.

According to Woo *et al*:

In an n-task A-agent concurrency, the total amount of communication C is

$$C = A \log_2 A \qquad (3)$$

And the total number of cycles S is:

$$S = n/A + 3 \log_2 A \qquad (4)$$

The amount of communication per agent per cycle C/A/S, in an n-task A-agent concurrency, is:

$$C/A/S = \frac{\log_2 A}{(n/A) + 3 \log_2 A} \qquad (5)$$

For a given task size, n = const., the amount of communication per agent per cycle approaches a constant, as the number of agents A increases:

$$\lim_{A \to \infty} C/A/S \,|_{n=\text{const.}} = 1/3 \qquad (6)$$

In an n-task A-agent concurrency, the average amount of processing approaches zero, as the number of agents increases; at the same time, idleness complements communication:

$$\lim_{A \to \infty} I/A/S = (1 - C/A/S) \qquad (7)$$

In a large organization while A is not infinite, it clearly is considerable. Hence the need to improve communication is vital for concurrency to effect meaningful results. Otherwise all the time is taken up in inter-agent communication and processing diminishes toward zero.

5. SIMULTANEOUS MULTILATERAL COMMUNICATION

The old style of communication between Engineering and Manufacturing was unilateral: from Engineering to Manufacturing and back. There could be many iterations of this, for each instance of design change for each part. This is very similar to the "dash for the flames" approach to putting out a fire. Concurrent Engineering seeks to replace this with more of a bucket brigade approach: coordinated multilateral communication. Now when we move from delivering buckets of water to delivering highly complex manufactured goods we have more hand-offs to consider. The question becomes what are the most critical and how can these be accomplished?

From (6) and (7) above we see that communications involving many agents are critical. In general, these involve the hand-offs from Engineering to Manufacturing And the key enabling technology is 3D communications or ***CAD Conferencing***.

6. ENGINEERING/MANUFACTURING PROCESS

The evolution of the design/manufacturing process illustrates this point. Years ago, engineers collaborating on a project worked under the same roof--and often in the same room. Achieving consensus or clarifying a hard-to-understand design problem was done "bidirectionally"--that is, in the give-and-take of a face-to-face meeting. Today, however, members of a design/manufacturing team may work in different continents. Moreover, while formal design reviews still play a critical role in developing new products, informal design reviews--those between a designer and a manufacturing engineer performing manufacturability studies, for example--are becoming increasingly important to achieving key business goals, such as decreasing time to market and lowering product costs. Conversely, companies not providing a technology infrastructure that supports these informal design reviews often inadvertently introduce bottlenecks, making it that much harder for engineers to achieve these goals.

To some extent, the problem of communicating design intent has been addressed by the increasing adoption of computer-aided-design programs based on three-dimensional solid or surface modeling. However, the most significant development that promises to bridge the communications gap is the rise of the Internet. The Internet is dramatically improving communications between all the parties bringing a new product to market. Now, a new generation of communications software is becoming available that enables engineers to transmit information contained in a 3-D product model to anyone with access to the Internet. With the aid of an inexpensive and easy-to-use World Wide Web browser, recipients can generate virtually any information that they need from the model, such as section cuts and dimensions, using their desktop Pcs.

For instance, 2-D detaining typically is a time-consuming step, and reducing or eliminating it can dramatically reduce engineering costs and lead times. It also helps prevent errors because it addresses the problem of mis-communication during the detailing process.

The engineering-change process can also be streamlined by using such tools because they make it much easier to update everyone on a project team. When paper drawings are distributed, it's often not clear which set of drawings are the most up-to-date, and the process itself is complex, tedious, and wasteful of time and money.

The most compelling reason to eliminate detailing, however, is the improved communication it affords between design and manufacturing. Replacing inherently ambiguous 2-D drawings with a 3-D product model greatly improves the odds that the designers' intent will be understood the first time. (Even if there are still questions, it's still more efficient for recipients to e-mail their questions--along with a 3-D attachment--to clarify the matter than it is for them to use traditional means.) Comprehension is further aided by the ability of manufacturing staff to generate meaningful information, especially when one considers that these employees in the past depended on detailers--who may work at another site--to anticipate their needs ahead of time.

Finally, e-mail and CAD conferencing are bi-directional, and therefore provide a computing infrastructure that makes it easier for manufacturing engineers to influence the design process--and, in the long term, to reduce time to market, shrink cycle times, and slash product costs.

This is significant because it reduces the impact of one barrier to concurrent engineering: the need to document every imaginable detail that some downstream user might need to know. Instead, end users can leverage the inherent superiority of 3D product models by extracting just the information they need and producing custom documents on the fly.

8. CONCLUSION

Using the results of Woo *et al*, we have attempted to show that the hand-off between Engineering and Manufacturing is critical to the Agile enterprise. And we have argued that the most efficient enabling technology to accomplish this efficiently is CAD Conferencing, whereby each individual is capable of customizing the information to his or her needs.

9. REFERENCES

1. [W95] T. C. Woo, C. C. Hsieh, Y. C. Fang, M. E. Wang, "Industrial Engineering", Technical Report IE-95-9-1, University of Washington, Seattle, WA, 98195-2650, USA,.
2. [S48] C. E. Shannon, "A mathematical theory of communication", Bell System Technical Journal, 27:379-423, 623-656

COMPUTER-AIDED PARTING LINE GENERATION FOR FREE-FORM SOLID MODELS

T. Wong, S. T. Tan and W. S. Sze
Department of Mechanical Engineering
The University of Hong Kong
Hong Kong

ABSTRACT

The parting line of a product model determines the layout of the mould block. The presence of free-form surfaces and interlocking regions increase the difficulties for computer-aided parting line/surface formation algorithm to determine the parting line and parting surfaces. Previously proposed algorithms are mainly dealing with products with planar or simple curved surfaces. A new algorithm, which locates the parting line and parting surface of a product by recursive slicing operation, is proposed. With the proposed algorithm, the parting line and parting surface of a free-form solid model can be located quickly and effectively.

KEYWORDS

Free-form solid, Model, Slicing, Grid Line Test, Column Intersection Test

1. INTRODUCTION

Traditionally, the mould design process relies heavily on the experience of the mould designer. The most time consuming parting line determination process is performed by the mould designer/maker. In determining a parting line, several criteria are considered. The criteria includes: the flatness of the parting surface, the mould opening distance during ejecting cycle, the shrinkage ratio, the pressure inside the mould cavity during injection cycle, cooling efficiency and the most important one, i.e. to relieve any unavoidable undercut. A mould designer can easily locate the parting line for a simple product model containing few undercuts (i.e interlocking regions). If the product model is complex in shape or has a number of undercuts, the mould designer may find it difficult to determine the parting line and parting surface.

An effective computer-aided parting line/surface formation algorithm can increase the overall efficiency in the mould design and manufacturing processes. Up to now, very few parting line formation algorithms has been reported. However the benchmark reported in these algorithms are very slow in terms of CPU times. In this paper, a simple but effective algorithm to obtain the parting line and parting surface by slicing the product model is proposed. The algorithm can quickly locate several parting lines and detect the presence of any undercut of a product model for a selected draw direction.

2. RELATED WORK

In a mould setup, the parting line is a line produced on the product, formed as the junction of the parting surfaces [1]. Pye [2] suggested that the parting line should be around the position of maximum dimension of a product when viewed in the draw direction of an injection moulding process. The criteria for choosing a suitable parting line may vary from case to case. Ravi and Srinivasan [3] proposed nine criteria for computer-aided parting line and parting surface design. These criteria can help a user to decide a most suitable parting surface amongst several alternatives. The criteria are: 1) maximum projected area, 2) maximum degree of flatness of the parting surface, 3) minimum draw distance, 4) presence of draft for easy withdrawal of moulded parts, 5) minimum number of undercuts, 6) maximum dimensional stability, 7) minimum volume of flash, 8) location of surfaces to be machined, and 9) controlling feeders and directional solidification.

A number of algorithms had been proposed for generating possible parting surfaces. Ganter and Tuss [4] proposed a computer-assisted algorithm in parting line development for cast pattern production. By sectioning the product model at a user selected parting plane, a planar parting line can be obtained.

Ravi and Srinivasan also proposed two methods [5] for parting line generation. The sectioning method, similar to Ganter's, locates the parting line by sectioning the product model to obtain a planar parting surface. The silhouette method sweeps the projection of the product model along the draw direction towards the product model to obtain the parting surface.

Parting line generation by a triangular sub-division of the product model's surfaces was proposed by Kwong [6]. In the method, a draw direction is selected first. Then, the algorithm triangulates the surfaces of the product model and classifies the triangulated surface patches into visible faces, invisible faces and degenerate faces. The edges on the product model separating the visible faces from the invisible faces are the tentative parting edges. By collecting and suitably arranging these edges, the possible parting lines can be found.

Hui [7] proposed an algorithm which used a solid sweep operation to establish the mould halves. The algorithm constructs a heuristic function which estimates the amount of undercuts involved and assesses the suitability of choosing a particular direction as the main or side-core direction. A search over the set of possible directions is started in order to located the most suitable parting direction. The required two mould halves are formed by suitable sweep and subtraction operations between the cavity block and the core block.

The above algorithms determine the parting lines of product models having planar surfaces and simple curved surfaces. For product models containing complex free-form surfaces, the applicability and the efficiency of these algorithms are questionable. For example, in evaluating the parting line, most algorithms mentioned above require the surface normals of all surfaces on the product model. For a product model having free-form surfaces, the surface normal of each free-form surface is a continuous function. If a sub-division method is used to divide the free-form surface into facets, the amount of data involved would be considerable, leading to an increase in computational time and decrease in overall efficiency. A slicing algorithm proposed in this paper is much simpler in determining the parting line and parting surfaces of complex free-form objects.

3. SLICING ALGORITHM

The algorithm uses a slicing strategy to locate the parting lines of a product model along a draw direction. A bounding box enclosing the product model is evaluated to determine the required size for slicing. Then, a slicing direction is selected. In this algorithm, the three principal axes (X, Y and Z-axis) are selected as the slicing directions because most of the reasonable parting lines and parting surfaces of a product model are normally found in the principal axis directions. Due to this reason and for simplicity, the principal axes are chosen in our proposed algorithm. The algorithm then "slices" through the product model by a set of slicing planes with normals perpendicular to the draw direction. If Z-axis is chosen to be the draw direction, then X or Y-axis may be chosen as the slicing directions. The intersection curves between each surface of the product model and a "slicing" plane are computed. These intersection curves on each slicing plane are collected and joined to form one or more intersection loops (Figure 1).

For a given draw direction, the outer parting line should enclose the maximum projected area of the product model. Therefore, the left-most and the right-most points of the cross-sectional profile of the product model on each slicing plane should lie on the possible parting line for the selected draw direction (Figure 1). In order to reduce the data required in finding the extreme point but without the loss of accuracy, the intersection curves on each slicing plane are classified into two categories: 1) straight lines and 2) curves including arcs, conics and spline curves. The two end points of each straight line, the two end points and the points of inflection on each curve are extracted. The reason is that for the straight line, only the two end points may constitute the extreme points whereas for the curve, the two end points and points of inflection may also constitute the extreme points (Figure 2). By comparing the relative positions of these extracted points, the left-most and the right-most points are obtained. For those slicing planes having more than one intersection loops on a slicing plane, the corresponding left-most and right-most points on each loop are located. By grouping all the left-most and the right-most points into a list and arranging them in an order, say from left to right, the points between the first and the last member of the list are the points where the inner parting line(s) will pass through (Figure 3). Thus both the outer and inner parting lines can be obtained.

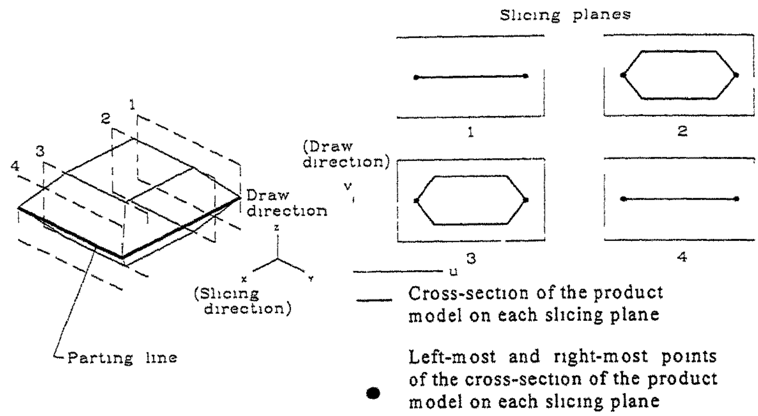


Figure 1 : The intersection profiles and extreme point on each slicing plane.

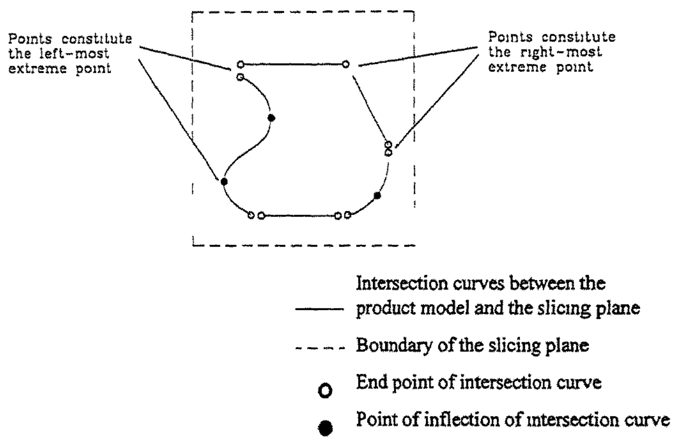


Figure 2 : Extracting points from each intersection curve on a slicing plane.

A product model may not have a unique parting line for a selected draw direction. This means that on a slicing plane, there exists more than one point on the left-most side, or the right-most side or on both sides of the cross-sectional profile. Taking Figure 4 as an example, the slicing plane encounters infinite number of extreme (left-most and right-most) points. In such situation, two sets of extreme points are recorded. These are the top extreme point and the bottom extreme point. An additional set of middle extreme point is recorded (by taking the mid-point between the top and the bottom extreme points) if the top and the bottom extreme points lie on the same straight line segment of the cross-sectional profile. If the top and the bottom extreme points are not on the same straight line segment, a recess usually exists. The right hand side of the intersection profile in Figure 4 illustrates this situation. The recess constitutes an undercut to the selected draw direction. In this case, the two outermost points of the recess are recorded as the middle extreme point (Figure 4). The same process is applied to each of the slicing planes.

By joining the corresponding extreme points on consecutive slicing planes, the outer and inner parting lines for a draw direction are obtained. For the case of a middle parting line shown in Figure 5, there are two extreme points on the right middle side, the parting line is split into ab and ac. The process continues until the split parting lines are joined again. The best one among the top, the middle and the bottom parting lines is chosen for the selected draw direction. The selection criteria are based on those proposed by Ravi and Srinivasan [3]. All the parting lines for the three principal axes as draw directions can be obtained by slicing the product model in any two of the three axes. This is because the first set of slicing operation, say along X-axis, gives rise to two sets of

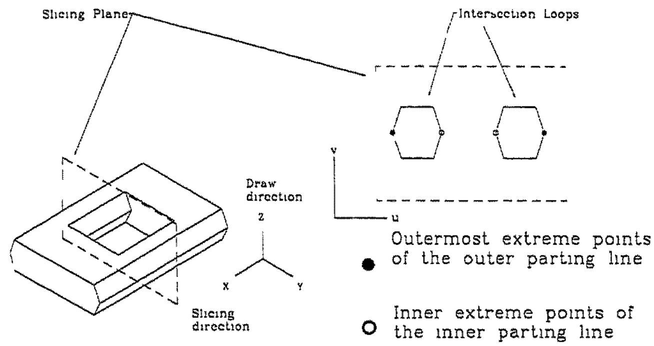


Figure 3 : Slicing plane and the outer and inner extreme points.

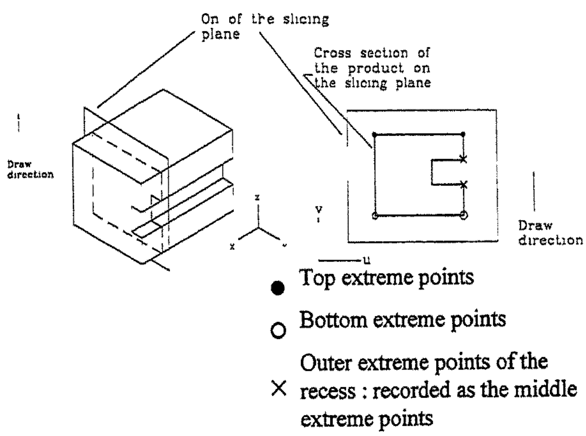


Figure 4 : Slicing plane with more than on extreme point on each side.

parting lines. As shown in Figure 6, on each slicing plane the extreme points along the horizontal and the vertical directions corresponding to the Z-axis and Y-axis as the draw directions respectively, can be computed simultaneously. Therefore, the second set of slicing operation along the Y-axis gives the extreme points on each slicing plane along the vertical direction with the X-axis as the draw direction (Figure 7). The main advantage of this algorithm is that it reduces the computation problem from a three dimensional Euclidean space into a two dimensional parametric space. This is because the necessary information of the parting line, even a non-planar one, can be extracted from the intersection profiles on a set of two dimensional slicing planes instead of extracting the information directly from the product model in the three dimensional Euclidean space.

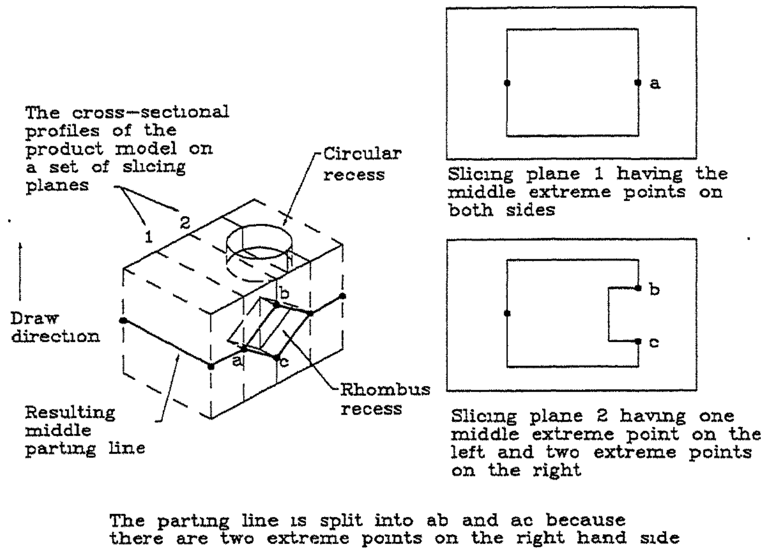


Figure 5 : Splitting and joining the middle parting line.

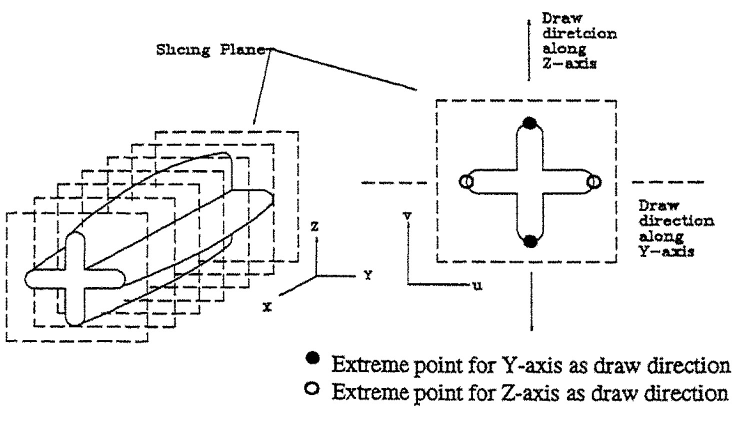


Figure 6 : First set of slices along a principal axis (X-axis), gives extreme points in two draw directions (Z and Y directions).

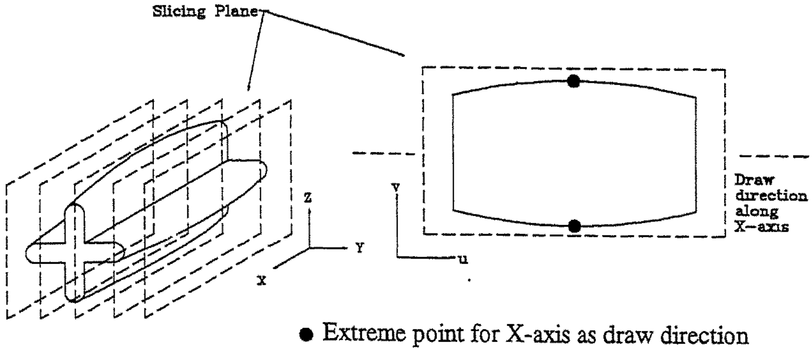


Figure 7 : Second set of slice along another principal axis (Y-axis), gives extreme points in the draw direction along the X-axis.

4. UNDERCUT DETECTION

Undercut is the interlocking region on the product model which prevents the simple ejection of the product from the mould for a selected draw direction. The relief of any undercut is one of the major criteria in choosing the best parting line among several candidates. In this paper, the undercut detection algorithm contains two tests: 1) a grid line test and 2) a column intersection test.

In Figure 8, a set of grid lines parallel to the draw direction is imposed onto a slicing plane. If the cross-section of the product model cuts a grid line in more than two points, like grid line number 6 and 7 in Figure 8, possible undercut surfaces are detected. After all the slicing planes are tested, a set of possible undercut surfaces can be located. This method reduces a lot of computational time because not all the surfaces of the product model are required for a thorough testing process. Again, the grid line test involves only intersection test on the two dimensional parametric space instead of the three dimensional Euclidean space.

Subsequent to the grid line test, a column intersection test is applied to confirm the suspected undercut surfaces. For a suspected surface, "walls" are extruded along the suspected surface's boundaries, with the extrusion direction being parallel to the draw direction (Figure 9). For each wall surface, an intersection test with other surfaces of the product model is performed. The directions of extrusion of the "walls" are in two directions i.e. along and opposite to the draw direction. The direction along the draw direction is tested first. If there is an intersection between the wall's surface and the product model's surfaces, then the opposite direction would be used to form the column and perform the test again. The suspected undercut surface is confirmed only when intersections between the column walls and the product model's surfaces were found on both directions.

After all the undercuts for a particular draw direction are located, the total volume of undercuts, total number of undercuts, position of each undercut relative to the lateral surfaces of a mould block, size of each undercut and the depth of each undercut can be calculated. By comparing the undercut information on each draw direction, the one with minimum undercuts and minimum size of the side cores to relieve the undercuts is determined. Together with other criteria stated earlier, the most suitable parting line among the three possible draw directions is obtained.

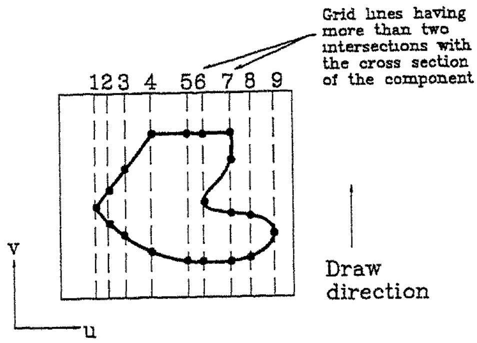


Figure 8 : The grid line test on each slicing plane.

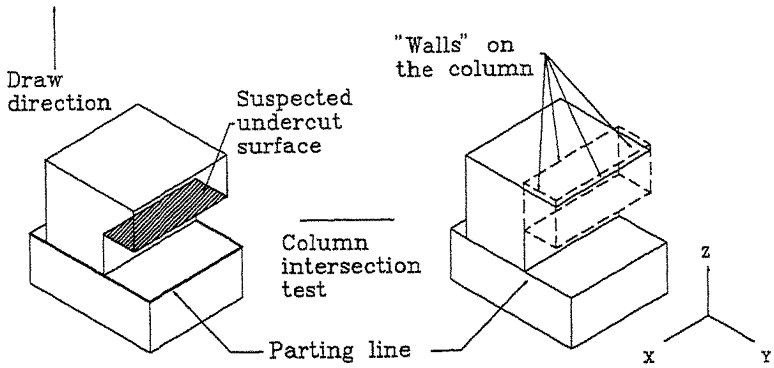


Figure 9 : The column intersection test on suspected undercut surface

5. RESULTS AND DISCUSSION

The major problem encountered in the algorithm was the accuracy in joining parting line between consecutive slicing planes. For example, there may be a "step" between two consecutive slicing planes. If the algorithm joins the corresponding extreme points directly regardless of the "step" between the slicing planes, the parting line may not accurately follow the profile of the product model. To overcome this problem, some checking criteria are imposed into the algorithm. The extreme point on each slicing plane must lie on one, or at most two intersection curves. As the intersection curve is resulted from the intersection between the slicing plane and the surface of the product model, the surface from which the intersection curve is derived can be found. By comparing the surfaces and the relative positions between corresponding extreme points, the existence of "steps" can be detected. In Figure 10, the right extreme point 1 is shared by three surfaces, A, B and C, whereas the right extreme point 2 is shared by surface B and D. Because not all the surfaces sharing the consecutive extreme point are the same, the joining process will trace the step profile. This resulted in a more accurate parting line to be determined.

The proposed algorithm has been implemented within the UNIGRAPHICS system running on a SGI Indigo2 workstation. Two testing pieces are shown below to illustrate the efficiency of the algorithm. They are 1) a fastening joint, 2) a track ball. The results are listed on the following table :

Testing components	# of faces	# of slicing planes	CPU time (second)
track ball	20	54	86 s
fastening joint	37	33	75 s

The results for both test pieces are quite good. The larger CPU time for the track ball is due to the presence of more complicated free-form surfaces. It is believed that the good performance of CPU time is mainly due to the fact that most of the computational processes were performed in two dimensional parametric space instead of three dimensional Euclidean space.

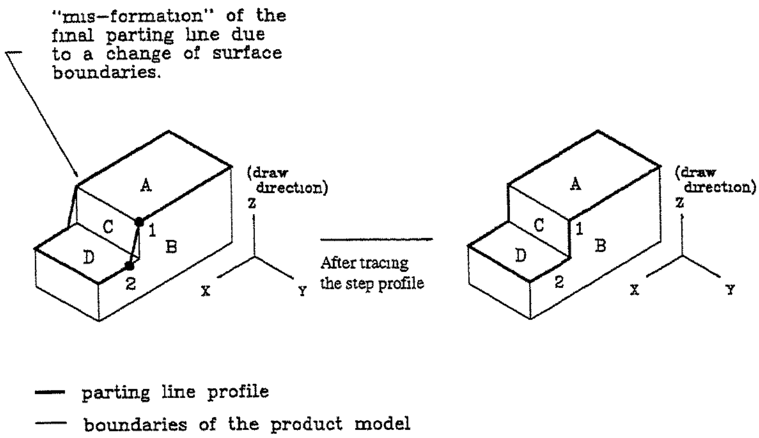


Figure 10 : The trace of the parting line at change of surface boundaries.

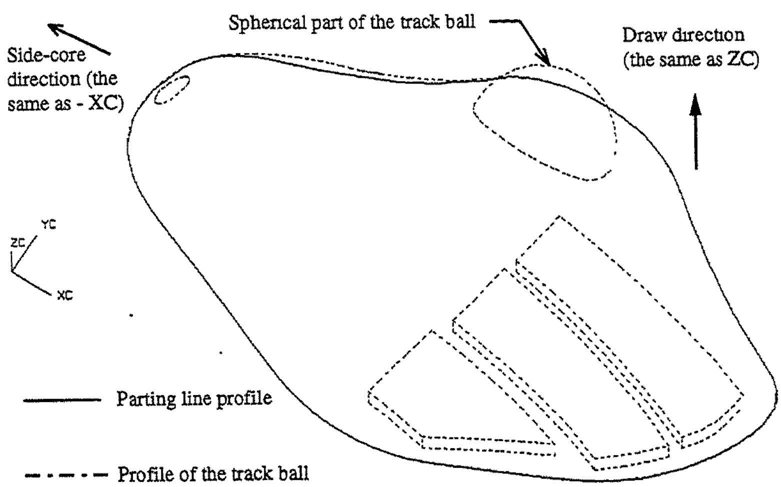


Figure 11 : The parting line for the track ball.

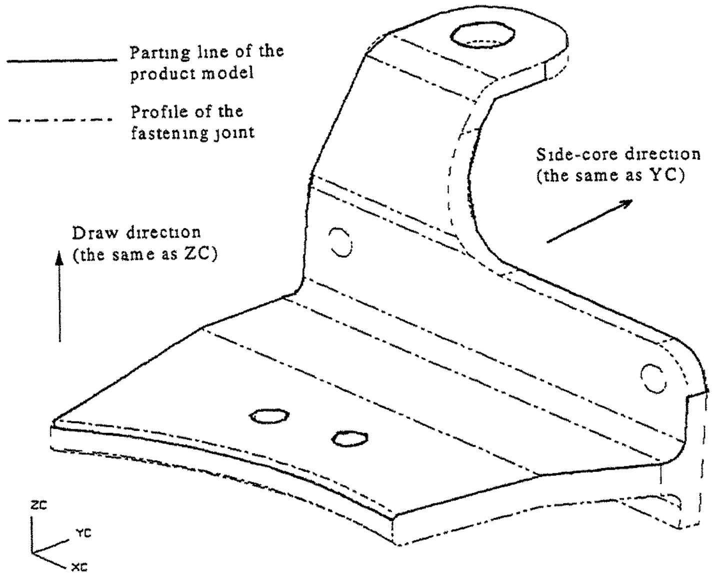


Figure 12 : The parting line for the fastening joint

6. ACKNOWLEDGMENTS

This research project is financially supported by the University Grant Council of the Hong Kong Government to which the authors are most grateful. Thanks are also due to the Department of Mechanical Engineering, The University of Hong Kong for providing computing and other testing facilities.

7. REFERENCES

1 PYE, R.G.W., Injection Mould Design: a design manual for the thermoplastic industry, p435, The Plastic and Rubber Institute, 1978.

2. PYE, R.G.W., Injection Mould Design: a design manual for the thermoplastic industry, p148, The Plastic and Rubber Institute, 1978.

3. Ravi, B. and Srinivasan, M.N., "Decision Criteria for Computer-Aided Parting Surface Design", Computer-Aided Design, Vol.22, No.1, Jan/Feb, pp.11-18, 1990.

4. Ganter, M.A. and Tuss, L.L., "Computer-Assisted Parting Line Development for Cast Pattern Production" Transaction of the American Foundrymen's Society, pp. 759-800, 1990.

5. Ravi, B. and Srinivasan, M.N., "Decision Criteria for Computer-Aided Parting Surface Generation", Proceedings, Manufacturing International Conference, Atlanta, pp.125-129, ASME, 1990.

6. Kwong, K.W., "Computer-Aided Parting Line and Parting Surface Generation in Mould Design", Ph.D. Dissertation, The University of Hong Kong, 1992.

7. Hui, K.C. and Tan, S.T., "Mould Design with Sweep Operations - A Heuristic Search Approach" Computer-Aided Design, Vol.24, No.2, Feb, pp. 81-91, 1992.

MODELLING CLOTH DEFORMATION

*Y.F. Zhao, *S.T. Tan, **T.N. Wong and +W.J. Chen
*Department of Mechanical Engineering, The University of Hong Kong, Hong Kong
**Department of Industrial and Manufacturing Systems Engineering, The University of Hong Kong, Hong Kong
+The Research Institute of Engineering Mechanics, Dalian University of Technology, Dalian, China

ABSTRACT

A constrained finite element method for modelling cloth deformation is developed. A geometric constraint according to the developable surface of the cloth object is introduced into the geometrically nonlinear finite element method (GNFEM). The membrane locking and solution distorting problems caused by large rotation and small internal forces are solved. The approximate solutions of cloth deformations are obtained by the constrained finite element method. The effectiveness of the proposed method is verified by comparing the approximate solutions with exact solutions. An example is given to show that the proposed method converges quickly.

KEYWORDS

Cloth Deformation, Large Deformation, Developable Surface, Constrained Finite Element.

1. INTRODUCTION

There is an increasing need for Computer-Aided Design (CAD) systems that can accurately model flexible cloth structures. In this paper, a new method for modelling cloth deformation is presented. The method can reproduce the behaviours of cloth deformation including draping deformation, hanging deformation, flowing deformation and etc..

Modelling cloth deformation is very difficult because the deformation is very complex. In cloth deformations, the tension stiffness is much stronger than the bending stiffness and the deformation is very large under gravity. This characteristic leads to very large rotations and displacements of cloth objects with small internal forces. The deformed surfaces of cloth objects can be assumed as developable surfaces. This characteristic of cloth deformation differs from that of the other materials. Cloth deformation is recoverable and repeatable, therefore, it is a small-strain/large-deformation problem which makes mechanical equations for solving the deformation become ill-conditional and strongly nonlinear. Besides, it is difficult to effectively model cloth deformation with the conventional mechanical methods for nonlinear problems such as the geometrically nonlinear finite element method (GNFEM) This is due to the problems of membrane locking and solution distorting involve in the conventional nonlinear mechanical methods.

A more radical aim of analyzing cloth deformation is to determine the characteristics and geometry after deformation. Thus, it may be advantageous to develop a special equation for modelling cloth deformation based on the mechanical properties and constraint conditions of cloth instead of applying or extending the conventional GNFEM. In this paper, the developable surface of the deformed cloth is regarded as a constraint condition and is introduced into the geometrical nonlinear finite element equations, and a constrained finite element method is established to model cloth deformation. The approximated solution of satisfying the mechanical properties and the constraint condition of developable surface is obtained by the constrained finite element equations. The membrane locking and solution distorting problems can be solved. The effectiveness of the method is verified by comparing the approximated solutions with exact solutions. An example is given to show that the proposed method converges quickly and is thus computationally efficient.

2. LITERATURE REVIEW

In recent years, a great deal of effort has been made towards modelling the complex deformation of cloth, with the methods falling into three main categories: geometric methods, physical methods and physical-geometrical methods.

2.1 Geometric Methods

Geometric methods are based on the computer graphics techniques to model cloth deformation. The deformable surface of cloth object is simply constructed from a given 3D shape which may be represented by geometric equations and then mapping it with the texture of the cloth. Weil[1] introduced the pioneering work to approximate a hanging cloth as a 2D grid of 3D points. His method uses a two-step process to model a rectangular cloth structure hanging from several constraint points. The first step is to approximate the cloth surface using catenary curves within the convex hull of constraint points. The second step uses a relaxation technique to adjust the displacements between all grid points in order to achieve a reasonable cloth-like surface.

Hinds et al.[2] presented a system for the interactive design of garments. In their system, garments are considered as surfaces offset from the human body.

Van West et al.[3] reported a graphical simulation to describe the draping deformation of bi-directional fabrics over arbitrary surfaces. This method predicts constrained thread paths at all points on surface draped with a bi-directional fabric. The location of two crossing yarns on the 3D surface is identified. Fabric deformation, as with thread orientation, is controlled by the placement of the constrained threads on the surface to be draped.

Using these models, the cloth deformation can be easily approximated. However, as geometric methods, their models do not include any mechanical analysis and do not consider the properties of cloth materials. Geometric methods are capable of generating the simplest model for the appearance of cloth deformations, particularly wrinkles and fords, with a minimum number of computations.

2.2 Physical Methods

In the physical methods, the material properties are considered and mechanical analyses may be used. The mechanical properties are mainly described by standard parameters including Young's modulus, Poisson ratio, density and thickness. Based on the physical methods, Terzopoulos et al.[4] developed a physical-based model using elasticity theory to solved the general problem of modelling deformable objects (which include deformable curves, surfaces and solids). It is possible to model the statics and dynamics of a wide range of objects including string, rubber, cloth, paper, and flexible metals. The goal of Terzopoulos et al.'s work is to produce a model based on physical methods to create the motions for computer animation. The result is a simplified cloth model in which the elastic behaviour is assumed to be linear and the special properties of cloth material are missing. So the cloth object is simulated more like a thin rubber sheet. The model has been extended and implemented by Thalmanns' team [5,6,7] to simulate the animation of synthetic actors with a skirt blowing in a breeze. Another group[8] has used Terzopoulos' model and to distribute forces on the whole cloth to determine the cloth deformation and to animate realistically the cloth motion in the air flow.

Aono[9] described a computer model, also based on elasticity theory, to simulate the wrinkle propagation on cloth objects. A more conventional stress-strain energy method instead of the differential geometry metrics was used in his model. In order to simplify the complexity of general stress-strain relations for cloth, the effect of cloth shear strain was ignored and the cloth structure is considered as a linear elastic sheet. It can model successfully the cloth wrinkles by propagating wave of impulsion forces though the cloth. However, the model can not simulate the draping and hanging deformations of cloth objects because the deformations are nonlinear and it is necessary to consider the shear strain.

Amirbayat and Hearly [10] also described a strain energy model of a cloth sheet for studying the three-fold buckling of textile fabrics. Breen et al [11] developed a particle-based model of cloth draping behaviour which successfully simulates the draping behaviour of woven cloth of different materials.

It has been recognized that cloth material properties and nonlinear large deformation must be considered in studying cloth deformation. The finite element method (FEM) is commonly used to solve this kind of the problems. Lloyd[12] used FEM to handle the various nonlinearities exhibited by deformations of cloth materials and to simulate a series of experiments on cloth structures. Collier et al.[13] presented a finite element approach to model the draping behaviour of cloth. Plate elements are used to predict the deformations of cloth during draping as a small-strain/large-displacement problem in finite element analysis. As part of an apparel CAD system, Okabe et al.[14] applied the finite element method to model the process of transferring a 2D cloth pattern to a 3D garment. Chen and Govindaraj[15] used the shear flexible shell theory to predict the drape of fabrics. The fabric is defined at a continuous, orthotropic medium and is assumed to be a flat plate initially, which goes through large deformation during the draping process. The conventional GNFEM and a nine-node, degenerated shell element are used to calculate the draping deformation. Young's modulus, shear modulus, and Poisson's ratio are used in the model as anisotropic material properties. At the same time, Gan et al.[16] also used a GNFEM, similar to Chen et al.'s method, to analyze and model fabric deformation characterized by large displacements and rotations but small strains. They pointed out that fabrics have independent bending stiffness and tensile stiffness, bending deformation therefore dominates fabric deformation. The constitutive relation of fabrics mainly depends on their bending and shear properties. Eischen et al. [17] developed a software for the modelling and control of flexible fabric parts. Their software is based on geometrically nonlinear thin shell theory. Ascough et al. [18] presented a simple finite element model for cloth drape simulation based on the nonlinear geometrically finite element analysis. The simplest beam element is used to model cloth draping deformation in order to reduce the computational time.

The above FEMs contribute to prove that FEM can be used in computer aided modelling of cloth deformation and give a framework for further study. However, the application of FEM is still concentrated on investigating and determining the cloth material constants (the elastic constant matrices used in FEM) and using the conventional finite element equations. To model various cases of cloth deformation, the problems of time consuming, error accumulating and convergence (membrane locking) will exist in the modelling processes. The conventional methods, therefore, have not effectively solved problems involved the cloth deformation such as large nonlinear deformation with small strain.

2.3 Physical-Geometric Methods

Geometric-physical methods combine the physical and geometric approaches. When it becomes difficult to use the physical methods to solve a problem, geometric methods provide a simple way to approximate the solution. Rudomin [19] suggested to reduce the computation time required by conventional physical method by using a geometric approximation as a start condition. His model gives an approximate shape of a draping cloth by the convex hull of the cloth and then Terzopoulos' deformable model is used to refine the shape. However, it is quite time-consuming in finding the convex hull of a polyhedral object. Kunii and Gotoda[20] were also aware of the difficulty in using either the physical methods or the geometric methods to model cloth objects. A geometric-physical method was established in which local analysis is based on the physical method and global analysis is solved by differential geometry.

3. THE PROPOSED METHOD

The physical-geometric model proposed in this paper is based on GNFEM. The developable surface of cloth object, as a constraint condition, is introduced to the GNFEM. A constrained finite element method which is special for modelling cloth deformation is established.

3.1 The Geometrically Nonlinear Finite Element Method

A finite element study of the behaviours of cloth deformation can be undertaken by considering the deformation of cloth objects with certain material properties. The simplest related case of using FEM is one involving only small strain and small displacements. However, the cloth deformation is a small-strain/very large-displacement geometrically nonlinear problem. In GNFEM, the deformation process described by the incremental method of the equilibrium equation as

$$\sum_{i}^{m}\mathbf{K}^{e}\cdot\Delta\mathbf{q}=\sum_{i}^{m}\Delta\mathbf{P}^{e}. \tag{1}$$

where $\mathbf{K}^e$ is the element stiffness matrix, $\Delta\mathbf{q}^e$ is the incremental displacement matrix, $\Delta\mathbf{p}^e$ is the incremental load matrix for the element, and m is the total number of elements. The nonlinearity is handled by calculating the stiffness matrix $\mathbf{K}$ in an incremental step as a function of the displacement $\mathbf{q}$. As a result, the cloth deformation can be obtained by using the small-deformation formula in every incremental load step.

Although GNFEM is very useful in modelling and analyzing large deformation problems of engineering structures, the deformation of cloth object is much differed from the large deformations of general engineering structures. The cloth deformation under gravity leads to large rotation with small internal forces. It is difficult to solve the problems once the geometrically nonlinear finite element equations are directly used in modelling the deformation with small-strain/large rotation (displacement). For example, in dealing with the large rotation by using GNFEM, the large accumulating error not only results in the solution distortion of deformation, but also causes the membrane locking problem. The problems have been the difficulties of GNFEM until now[21].

In cloth deformation under own weight, its internal force is always small. Since the cloth surface is developable and there is no extension in the membrane direction, it can just control the internal force. Therefore, if the developability of cloth surface is introduced to control the internal force, the above difficult problems involved in using GNFEM will be solved. In the calculation of cloth deformation, the approximate solution of satisfying the developability of cloth surface and mechanical properties can be obtained.

3.2 The Constrained Finite Element Method

In modelling cloth deformation by the constrained finite element method, the process of controlling the cloth deformation is concluded that the cloth surface is kept developable during the deformation. In other words, the developable surface is only an acceptable cloth surface, and it is a basic condition or a geometric constraint. The assumption of the constraint is given that the length of all threads of a cloth object remain unchanged after the cloth deformation. That is

$$G=L_i-\ell_i=0 \qquad (i=1,2,3\cdots \mathrm{n}) \tag{2}$$

where n is the total number of threads, L_i and ℓ_i represent the lengths of initial and deformed threads respectively. Although the assumption has been presented by some previous works[3,9,12,22], this is the first attempt to regard it as a constraint and to import it into the governing equation of cloth deformation.

We can introduce this constraint by forming a new functional of controlling cloth deformation

$$\Pi^{*}=\sum_{e}^{m}\left(\Pi^{e}+\lambda\|G\|^{e}\right) \tag{3}$$

where $\|G\|^{e}=\sum_{i}^{n}G_i^2$ is the element constraint function and n represents the number of threads of a cloth element. By the constrained variational principles[23], we have

$$\delta\Pi^{*}=\sum_{e}^{m}\left(\delta\Pi^{e}+\lambda\delta\|G\|^{e}\right) \tag{4}$$

where the variation of the original functional Π^e has been obtained by GNFEM[23], and the detailed derivation of variation of the constraint function $\|G\|^e$ has been given in our other papers[24,25]. Thus, the governing equation of the constrained finite element is obtained as

$$\sum_{e}^{m}(\mathbf{K}^e + \lambda\mathbf{K}_c^e)\Delta\mathbf{q} = \sum_{e}^{m}[\Delta\mathbf{P}^e + \lambda\mathbf{R}_c^e] \tag{5}$$

After introducing the geometric constraint into large-deformation formula, it is found that problems of modelling cloth deformation can be simplified in using GNFEM. Because of the constraint introduced to control the internal forces, the membrane locking can be suppressed. The problem of solution distortion can also be avoided. The numerical analysis is given in the following section to verify the effectiveness of the constrained finite element methods.

4. Numerical Analysis

In this section, the comparison of results obtained by the constrained finite element method and GNFEM is first given. As can be seen in Case 1, the solution distorting problem involved in GNFEM (Figure 2) can be avoided by using the constrained finite element method (Figure 3). Then, since the exact solution of the hanging deformation of a thread is given by the catenary equation, Case 2 gives an evaluation of the constrained finite element method by comparing the exact solution with the approximate result of a thread deformation. As shown in Figure 4, the computational result of the thread deformation approximates well to the exact results.

4.1 Case 1

This is a basic case in which a rectangular cloth is fixed at four corners and hung freely in 3-D space. With the piece of cloth initially located at the z=0 plane, it is subdivided into elements as shown in Figure 1. By using GNFEM without the constraint condition, the deformed surface resulting from the hanging of the cloth is obtained as shown in Figure 2(a) which is a wire-frame display. Figure 2(b) is a development of the deformed surface and the surface area of the development of the deformed surface has changed from 100 to 118.33 cm^2. It results that the deformed surface is distorted and the constraint condition is not satisfied. Figure 3 shows the results of the same case obtained by using the constrained finite element method. Obviously, the solution distortion involved in large rotations has been effectively controlled, and the geometric constraint of cloth deformation is successfully satisfied. As can be seen in Figure 3(b), the area change of the deformed surface is limited and within 0.04%.

Although the constrained finite element method is successful in satisfying the geometric constraint as illustrated in Figure 3(b), the deformation cannot be verified by an exact solution. It would be more appropriate to scientifically evaluate a simulation against a previously acknowledged computational result and to obtain a more accurate evaluation. Since an exact solution using the catenary equation [26] exists in the case of the hanging deformation of a thread, it is used as the basis for evaluating our proposed method by comparing the exact solution with the approximated results of a thread deformation.

4.2. Case 2

A thread is hung between fixed end points A and B which are at the same level and distance of 5 cm apart. The length, width and thickness of the thread are 7.0cm, 0.1cm and 0.1cm respectively and the length is subdivided into 14 elements. The Young's modulus and the Poisson ratio are 12.0gk/cm^2 and 0.05. A comparison between the results obtained by using the constrained finite element method and those according to the catenary equation with the same input data is shown in Figure 4.

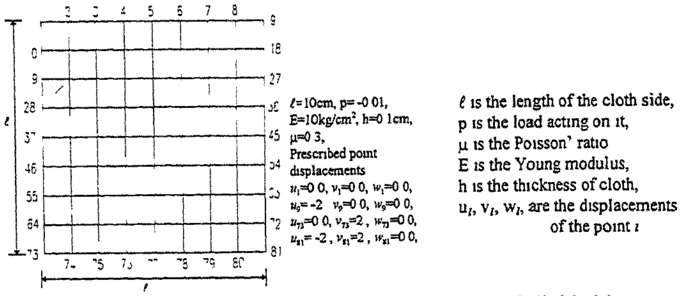
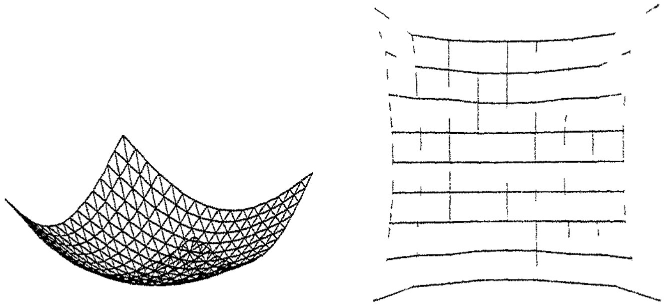
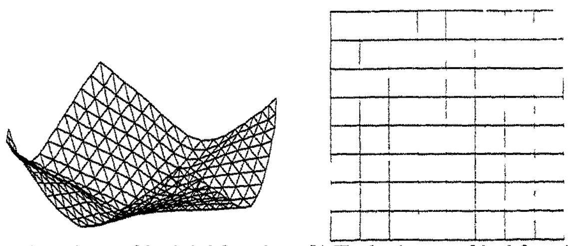


Figure 1 A piece of rectangular cloth subdivided into a mesh



(a) A wire frame display of the cloth deformation (b) The development of the deformed surface (a)

Figure 2 The computational result of cloth deformation by using GNFEM



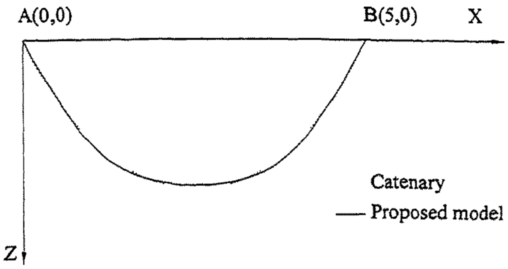
(a) A wire frame display of the cloth deformation (b) The development of the deformed surface (a)

Figure 3 The computational result of cloth deformation by using the constrained finite element method

5. Example

A 70cm × 70cm cloth object is draped over a 35cm × 35cm cube after being initially located on a 2D plane just above the cube. The thickness is 0.1cm and the Young's modulus and the Poisson ratio are 14.0gk/cm² and 0.03. the cloth is subdivided into 128 triangular elements. Using the proposed method, the computation for the deformation repeats until the cloth has draped over the cube and has obtained a geometric displacement of its equilibrium position. Wire-frame diagrams of the results of the cloth deformation are shown in Figure 5(a). Figure 5(b) shows the draping surface with a background after texture mapping.

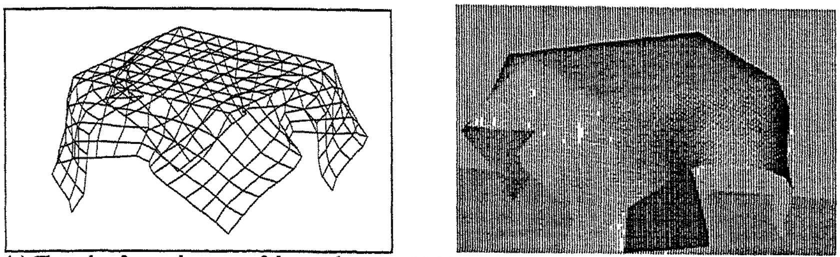
In this Example, the proposed model gives a reasonable solution for modelling cloth object by using the constrained finite element method. The development of the deformed surface approximates the area of the initial 2D surface within a 0.06% error margin. Furthermore, the method is stable and converges fairly quickly. This is illustrated in the above example where only 5 iterations are required, and the computational time used in the above example is under one minute on a DEC 3800s AXT computer.



	X		Z	
No	Catenary	Proposed model	Catenary	Proposed model
1	0 000	0 000	0 000	0 000
2	0 244	0 244	-0 464	-0 436
3	0 500	0 500	-0 874	-0 866
4	0 782	0 782	-1 247	-1 279
5	1 116	1 116	-1 595	-1 651
6	1 527	1 527	-1 904	-1 936
7	2 002	2 002	-2 116	-2 093
8	2 500	2 500	-2 189	-2 136
9	2 998	2 998	-2 116	-2 093
10	3 473	3 473	-1 904	-1 936
11	3 884	3 884	-1 595	-1 651
12	4 218	4 218	-1 247	-1 279
13	4 500	4 500	-0 874	-0 866
14	4 756	4 756	-0 464	-0 436
15	5 000	5 000	0 000	0 000
L^*	Catenary 7 00000		Proposed 7 00074	
Err	0 0%		0 01%	

L^* is the length of the thread after deformation

Figure 4 Comparison of results from the catenary equation and the proposed method.



(a).The wire-frame diagram of the result (b).The display of the result with texture and background

Figure 5. A cloth object is draped over a square table.

6. Conclusion

Modelling a cloth object requires a suitable method which considers not only the mechanical properties but also the geometric constraint of the cloth object. The advantages of the proposed method are:

- The proposed method is efficient in modelling cloth objects. The load is applied in a single step and less than 10 iterations are needed to model the cloth deformations such as draping and hanging deformations. The method is stable and converges fairly quickly.
- In the proposed method, large deformation can be achieved by introducing the geometric constraint into the conventional GNFEM and the difficult problems involved in GNFEM can be avoided.

6. ACKNOWLEDGMENTS

This work has been supported by the Department of Mechanical Engineering, the University of Hong Kong and a research grant funded by the Research Grants Council, Hong Kong.

7. REFERENCE

1. Weil J., "The Synthesis of Cloth Objects," Siggraph, Vol. 20, No. 4, pp. 49-54, 1986.
2. Hinds B.K., McCartney J. and Woods G., "Pattern Development for 3D Surfaces," Computer Aided Design, Vol. 23, No. 8, pp. 583-592, 1991.
3. Van West B. P., Pipes R. B. And Keefe M., "A Simulation of the Draping of Bidirectional Fabrics Over Arbitrary Surfaces", Journal of Textile Institute, Vol. 81, No. 4, pp. 448-460, 1990.
4. Terzopoulos D., "Elastically Deformable Models," Proc. Siggraph'87, Vol. 21, No. 4, pp. 205-241, 1987.
5. Thalmann N. M. and Thalmann D., "Complex Models for Animating Synthetic Actors", IEEE Computer Graphics and applications, Vol. 11, No. 5, pp. 32-44, 1991.

6. Volio P., Thalmann N. M., Shen J. and Thalmann D., "An Evolving System for Simulating clothes on Virtual Actors", Computer Graphics & Applications, Vol. 5, No. 9, pp. 42-51, 1996.
7. Yang Y.and Thalmann N. M., "An Improved Algorithm for Collision Detection in Cloth animation With Human Body", Proc. Of PacificGraphics'93, Vol. 1, pp. 237-251, 1993.
8. Li L. Damodaran M. And Gay R. K. L., "Aerodynamic Force Models for Animating Cloth Motion in Air Flow", Visual Computer, Vol. 12, pp. 84-104, 1996.
9. Aono M., "A Wrinkle Propagation Model for Cloth", CG Internation'90 Springer Verlag, pp. 95-115, 1990.
10. Amirbayat J. and Hearle J. W. S., "The Complex Buckling of Flexible Sheet Materials-Part I. Theoretical Approach", J. Mechanical Science. Vol. 28, No. 6, pp. 339-358, 1986.
11. Breen D. E., House D. H., and Getto P. H., "A Physically-Based Particle Model of Woven Cloth," The Visual Computer, Vol. 8, No. 8, pp. 264-277, 1992.
12. Llord D. W., "The Analysis of Complex Fabric Deformations", Mechanics of Flexible Fibre Assemblies, eds. Hearle J. W. S., Thwaites J. J. And Amirbayat J., pp. 311-342, 1980.
13. Collier J. R, Collier B. J., O'Toole G. and Sargand S. M., "Drape Prediction by Means of Finite-element Analysis", J. Textile Institute, Vol. 82, No. 1, pp. 96-107, 1991.
14. Okabe H. Imaoka H. Tomiha T. and Niwyo H., "Three Dimensional Apparel CAD System," Proc. Siggraph'92, Vol. 26, No. 2, pp. 105-110, 1992.
15. Chen B. And Govindarj M., "A Physically Based Model of Fabric Drape Using Flexible Shell Theory", Textile Research Journal, Vol. 65, No. 6, pp. 324-330, 1996.
16. Gan L. And Ly N. G., "A Study of Fabric Deformation Using Nonlinear Finite Elements", Textile Research Journal, Vol 65, No. 11, pp. 660-668, 1995.
17. Eischen J. W. Deng S. And Claap T. G., "Finite-Element Modeling and Control of Flexible Fabric Parts", IEEE Computer Graphics and Applications, Vol. 5, No. 9, pp. 71-80, 1996.
18. Ascough J., Bez H. E. And Bricis A. M., "A Simple Finite Element Model for Cloth Drape Simulation", International J. of Clothing Science and Technology,Vol. 8, No. 3, pp. 59-74, 1996.
19. Rudomin I. J., "Simulating Cloth Using a Mixed Geometric-physical method," Ph.D. Thesis, University of Pennsylvania, 1990.
20. Kunii T. L. And Gotoda H., "Modeling and Animation of Garment Wrinkle Formation Processes," Computer Animation'90, Springer-Verlag, pp. 131-147, 1990.
21. Heinrich Schoop, "Standard test for membrane locking of plate elements in geometrically nonlinear bending", Mechanicas Advance, Vol. 22, No. 2, pp. 284-287, 1992.
22. Zhao Y.F., Tan S.T. and Wong T.N., "An Effective Method for modelling Flexible Surfaces of Cloth Objects", Proc. of Advances in Design Automation 1994, ASME Vol.1, pp.351-358, 1994.
23. Zienkiewicz O.C. " The Finite Element Method", McGRAW-HILL Book Company Limited, 1977.
24. Zhao Y. F., Tan S. T., Wong T. N and. Chen W.J, "A Model for Simulating Flexible Surfaces of Cloth Objects", Comput. and Struct., Accepted, 1996.
25. Zhao Y. F., Tan S. T., Wong T. N and. Chen W.J, "Computer-Aided Modelling of Cloth Deformation", Proc. of Advances in Design Automation, ASME, 1996.
26. Irvine M., Cable Structures, Dover Publications, Inc., New York, 1992.

An Application of Spline Surface Interpolation on Blending/Polishing System for Turbine Blade

Wang Jia ye, Lee Eng Wah, Shi Cheng Fu, Liu Pei Ling
Gintic institute of Manufacturing Technology
Nanyang Technological University,
Singapore

ABSTRACT

One of the major processes in repairing turbine blades in the aerospace industry is blending. Blending is very labor intensive and demands high skill. Automation of this process is one of the technology upgrading programme of the aerospace industry.

Blades are rotated rapidly in a turbine, and the working temperature is very high. After a long working duration, some regions of the blades are corrupted and rubbed. The regions along the front edge and end edge of the blade are damaged seriously (ref. Figure1).

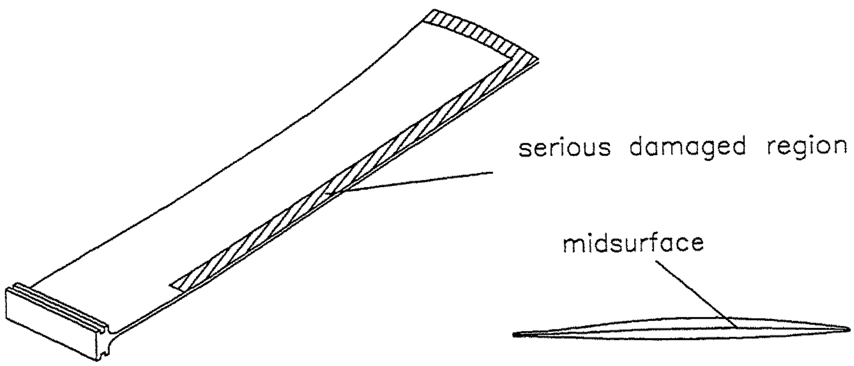


Figure 1

The blending process of repairing the blade is cutting the serious damaged regions, welding a piece of same metal of the blade, and then milling the attached metal to obtain an ideal shape of the blade. Since a blade is working in high stress, itself is also distorted after long time continuously working (ref. Figure2). Although to repair a blade is not to amend the distortion of the blade, the distortion is really make a trouble for maintaining the damaged region. Because of the distortion the milling NC data can not be generated from the original design. The blade can be considered an elastic plate with certain boundary conditions. According to mechanical principle an algorithm is presented to regenerate the surface of the blade.

KEYWORDS

Blade, B-spline, Interpolation.

ALGORITHM

The algorithm of regenerating the blade surface is base on the distorted shape and the original design. We will name the blade needed to repair as distorted blade. The ideal shape of the repaired blade is same as the distorted blade without damage.

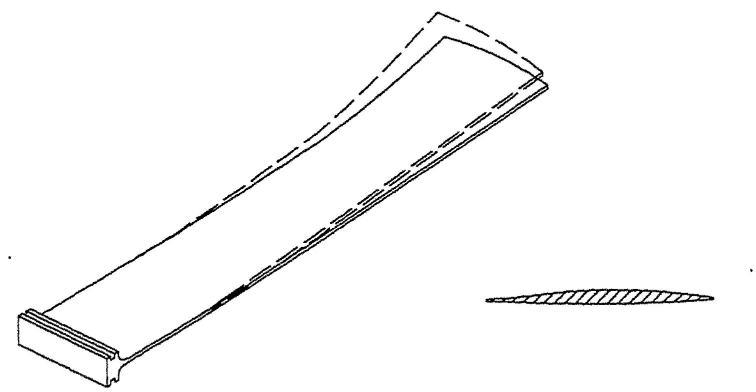


Figure 2

To solve this problem, firstly a complete midsurface of the distorted blade is generated. According to the deviation between the surface and the midsurface of the original blade design the repaired surface of the distorted blade can be generated. The midsurface of a blade can be considered an elastic plate. The distortion of the blade should obey the mechanical principle. Since spline surface has the background of a mechanical elastic plate, the interpolation of spline can be a good approximation for the midsurface of the distorted blade. To simplify the description, we assume that the projection of the blade on the xy plane is a rectangle(ref. Figure 3). Generally we will discuss B-spline parameter surface of u and v instead of x and y., and the blade midsurface corresponds to a B-spline parameter surface on a rectangle (0<v<v0, 0<u<u0).

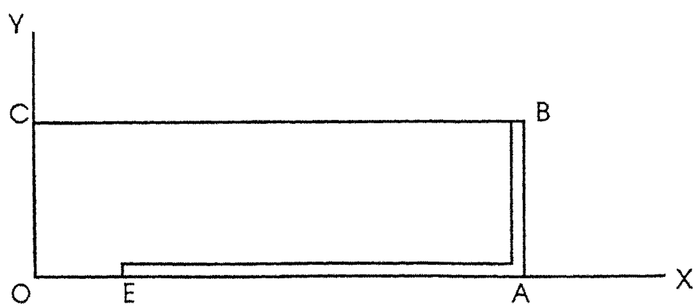


Figure 3

The most of the interpolation points of the midsurface can be obtained by measuring the surface of the damaged blade by 3D scanner and removing the part of the deviation between the surface and the midsurface of the original blade design. Only the serious damaged region can not be measured precisely. It is therefore impossible to interpolate the whole midsurface. Fortunately, the serious damaged regions are narrow. It is sufficient to obtain the interpolation midsurface by adding some conditions along the damaged sides EA and AB.

Since the blade working in a turbine with free boundary condition on the sides OA and AB, the boundary conditions on EA are

$$\frac{\partial^2 W}{\partial Y^2}=0, \qquad \frac{\partial^3 W}{\partial Y^3}=0. \tag{1}$$

The boundary conditions on BC are

$$\frac{\partial^2 W}{\partial X^2}=0, \qquad \frac{\partial^3 W}{\partial X^3}=0, \tag{2}$$

where W(X,Y) is the deflation of the midsurface.

Using the notation from Chapter 10 of [2] a tensor product B-spline surface may be written as

$$W(x,y)=\sum_{i=1}^{n-2}\sum_{j=1}^{m-2}P_{i,j}N_i^3(x)N_j^3(y), \tag{3}$$

where we assume that one knot sequence (xi i=0,1,2,....n+1) in the x direction and one (yj, j=0,1,2,,...m+1) in the y direction are given (Ref. Figure 4). The knot sequences are based on triple end knots. That is

$$x_0<=x_1=x_2=x_3<x_4<...<x_{n-3}<x_{n-2}=x_{n-1}=x_n<=x_{n+1} \tag{4}$$

and

$$y_0<=y_1=y_2=y_3<y_4<...<y_{m-3}<y_{m-2}=y_{m-1}=y_m<=y_{m+1}. \tag{5}$$

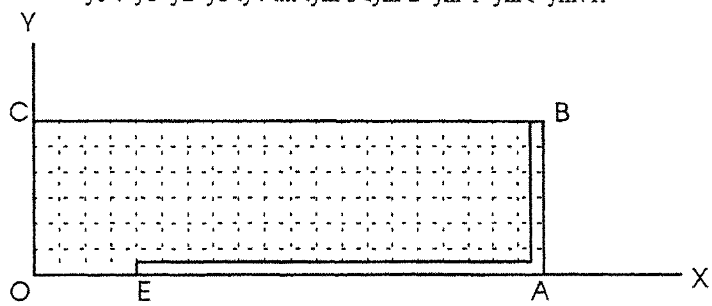


Figure 4

B-spline $N_l^k(x)$ can be calculated by the recursion formula:

$$N_l^k(x)=N_l^{k-1}(x)\frac{x-x_{l-1}}{x_{l+k-1}-x_{l-1}}+N_{l+1}^{k-1}(x)\frac{x_{l+k}-x}{x_{l+k}-x_l} \tag{6}$$

$N_l^0(x)$ is defined by

$$N_l^0(x)=1 \qquad \text{if } x_{l-1}<x<x_l$$

(7)

$$N_i^0(x) = 0 \qquad \text{else}$$

From (4), (6) and (7), it follows that

$$N_1^3(x) = \frac{(x_4 - x)^3}{(x_4 - x_1)^3} \tag{8}$$

$$N_2^3(x) = \frac{x - x_1}{x_4 - x_1}\left(\frac{(x_4 - x)^2}{(x_4 - x_1)^2} + \frac{(x_5 - x)(x_4 - x)}{(x_5 - x_1)(x_4 - x_1)} + \frac{(x_5 - x)^2}{(x_5 - x_1)^2}\right) \tag{9}$$

$$N_3^3(x) = \frac{(x - x_1)^2}{x_5 - x_1}\left(\frac{(x_4 - x)}{(x_4 - x_1)^2} + \frac{(x_5 - x)}{(x_5 - x_1)(x_4 - x_1)} + \frac{(x_6 - x)}{(x_6 - x_1)(x_4 - x_1)}\right) \tag{10}$$

$$N_4^3(x) = \frac{(x - x_1)^3}{(x_4 - x_1)(x_5 - x_1)(x_6 - x_1)} \tag{11}$$

Assume that

$$x_6 - x_5 = x_5 - x_4 = x_4 - x_3 = x_4 - x_2 = x_4 - x_1 \tag{12}$$

From (8), (9), (10) and (11), we obtain

$$\frac{d^2}{dx^2} N_1^3(x_1) = \frac{6}{(x_4 - x_1)^2}, \qquad \frac{d^2}{dx^2} N_2^3(x_1) = \frac{-9}{(x_4 - x_1)^2},$$

(13)

$$\frac{d^2}{dx^2} N_3^3(x_1) = \frac{3}{(x_4 - x_1)^2}, \qquad \frac{d^2}{dx^2} N_3^3(x_1) = 0.$$

$$\frac{d^3}{dx^3} N_1^3(x_1) = \frac{-6}{(x_4 - x_1)^3}, \qquad \frac{d^3}{dx^3} N_2^3(x_1) = \frac{10.5}{(x_4 - x_1)^3},$$

(14)

$$\frac{d^3}{dx^3} N_3^3(x_1) = \frac{-5.5}{(x_4 - x_1)^3}, \qquad \frac{d^3}{dx^3} N_4^3(x_1) = \frac{1}{(x_4 - x_1)^3}.$$

From (3), for every (x_s, y_t) (3<s<n-4, 3<t<m-4) a equation can be created

$$\sum_{i=1}^{n-2} \sum_{j=1}^{m-2} P_{i,j} N_i^3(x_s) N_j^3(y_t) = Z(x_s, y_t), \quad 3<s<n-2,\ 3<t<m-2, \tag{15}$$

where $Z(x_s, y_t)$ is the deflation of the midsurface measured by 3D scanner. If s=3, n-2 or t=3, m-2, (x_s, y_t) is on the boundary of the polygon OABC (Figure 4). Two conditions must be specified for those point located at boundary. If the point is not on AE and AB, the deflation $Z(x_s, y_t)$ of the midsurface can be measured form the distorted blade, and let

$$P_{1,t} = P_{2,t} = Z(x_3, y_t), \quad P_{n-3,t} = P_{n-2,t} = Z(x_{n-2}, y_t),$$

(16)

$$P_{s,1} = P_{s,2} = Z(x_s, y_3), \quad P_{s,m-3} = P_{s,m-2} = Z(x_s, y_{m-2}).$$

If (x_s, y_t) is on AE and AB, two equations consisting with (1) or (2) must be created. From (3), (13) and (14) if $(x_s, y_t)$is on EA, the equations are

$$6\sum_{i=1}^{n-2} P_{i,1} N_i^3(x_s) - 9\sum_{i=1}^{n-2} P_{i,2} N_i^3(x_s) + \sum_{i=1}^{n-2} P_{i,3} N_i^3(x_s) = 0$$

$$-6\sum_{i=1}^{n-2} P_{i,1} N_i^3(x_s) + 10.5\sum_{i=1}^{n-2} P_{i,2} N_i^3(x_s) - 5.5\sum_{i=1}^{n-2} P_{i,3} N_i^3(x_s) + \sum_{i=1}^{n-2} P_{i,4} N_i^3(x_s) = 0. \tag{17}$$

If (x_s, y_t) is on AB, they are

$$6\sum_{j=1}^{m-2} P_{n-2,j} N_j^3(y_t) - 9\sum_{j=1}^{m-2} P_{n-3,j} N_j^3(y_t) + \sum_{j=1}^{m-2} P_{n-4,j} N_j^3(y_t) = 0$$

$$-6\sum_{j=1}^{m-2} P_{n-2,j} N_j^3(y_t) + 10.5\sum_{j=1}^{m-2} P_{n-3,j} N_j^3(y_t) - 5.5\sum_{j=1}^{m-2} P_{n-4,j} N_j^3(y_t) + \sum_{j=1}^{m-2} P_{n-5,j} N_j^3(y_t) = 0. \tag{18}$$

The liner equations (15),(17) and (18), $\{P_{i,j}\}$ (i=1,2,3,...n-2, j=1,2,3...m-2) can be solved by elimination method. From (3) we obtain the complete midsurface W(x,y).

REFERENCE

[1] Friedel Hartmann, The Mathematical Foundation of Structural Mechanics, Spring-Verlag Berlin Heidelberg New York Tokyo 1985.

[2] Gerald Farin, Curves and Surfaces for Computer Aided Geometric Design, ACADEMIC PRESS INC, 1990

INTELLIGENT SYNTHESIS OF MULTIAGENT MANUFACTURING SYSTEMS

Edward Szczerbicki
Department of Mechanical Engineering
The University of Newcastle, Newcastle, NSW Australia

ABSTRACT

A three stage approach to the development of multiagent manufacturing systems is proposed involving functional decomposition, agent representation, and information flow integration (synthesis). The information flow synthesis stage is discussed in detail. A rule-based agent synthesis for construction of autonomous subsystems and multiagent planning systems is presented. The approach formalises the integration process that a system analyst solves in an ad hoc manner. The approach presented facilitates the discovery of new integrated solutions. It involves generation of a set of autonomous agents for building an overall system. Hierarchically structured subsystems are developed at different levels of abstraction that parallel the approach used by an expert system analyst.

KEYWORDS

Information Flow, Integration, Multiagent Systems

1. INTRODUCTION

Management, manufacturing, design, and planning systems are often viewed as sets of components (agents) supporting separate functions. Many organisations operate in this highly compartmentalised manner. It appears that the general direction of systems development in the future, however, is toward linking together function-specific agents into fully integrated system. An integrated system is a system that consists of agents that efficiently contribute to the task, functional behaviour, and performance of a system as a whole. It is believed that such an integration can be achieved through the flow of information (Biemans [1], Burch and Grudnitski [2]). "Integration", as used in this paper, should not be confused with integration at a physical level by means of computer networks or computer buses. Rather, the semantics of integration is addressed - the information that agents should share.

The development process of an integrated system involves three stages:

- Stage 1: functional decomposition,
- Stage 2: agent representation,
- Stage 3: information flow base synthesis of agents.

The functional decomposition stage aims at providing functions of the system that is developed. The agent representation process assigns agents to the functions. Finally, Stage 3 uses the representations obtained at Stage 2 and integrates them into the developed system. In this methodology synthesis is also used for creating subsystems/systems with hierarchical structure. Earlier works by the author (Szczerbicki [3,4,5]) develop the tools for structuring and evaluation of information flow in autonomous systems that can be used at the agent representation stage. The focus of this paper is on the synthesis problem.

2. THE STAGE OF SYNTHESIS

The approach proposed in this paper employs a system-theoretic hierarchical integration which provides a practical tool for reasoning with agents and recognises the importance of interaction in knowledge-based systems and intelligent systems (Bobrow [6], Albus [7]). It is based on a working hypothesis that it is possible to develop hierarchically structured integrated systems by reasoning with autonomous agents. As a result, the architecture and information flow at the integration stage takes into consideration the following assumptions:

- hierarchical representation of agents with various levels of detail is suited for modelling of integrated systems,
- any hierarchical model of an integrated system can be characterised by its agents and connections that were applied during the matching process.

Traditionally, system analysts have been solving the integration problem in an ad hoc manner. In this paper, a formal integration approach is presented which is suitable for computer implementation.

3. THE PRINCIPLES OF AGENT INTEGRATION

The following are the basic concepts associated with the integration used for creating hierarchically structured systems at different levels of abstraction:

(i) agents,
(ii) connections,
(iii) ports,
(iv) subsystems.

An *autonomous agent* is a group of elements (e.g. people, machines, robots, guided vehicles) that decides about its own informational requirements (autonomy) and that represents function(s) generated in the stage of functional decomposition. *Connections* and *ports* link, through the flow of information, the agents and subsystems. A *subsystem* is a set of connected agents.

3.1. Agents

A description of the agent includes the following:

(i) functions,
(ii) informational input,
(iii) informational output,
(iv) internal information structure.

In the function section of an agent description the characteristic functions are listed. The characteristic functions are those that have been generated during the functional decomposition stage and then chosen to be represented by the agent during the representation stage. Informational input and output refers to the variables describing the state of the external environment of an agent. For an input, only those variables are included the information about which is necessary to perform functions represented by an agent. For an output, only those variables are included that are affected by the functions specified for an agent. Internal information structure models an information flow inside an agent. It is defined during the agent representation process.

3.2. Agent Connections

Agents are matched using informational input and output. For example, in the domain of manufacturing systems, information may represent material availability, tool availability, machine availability, number of parts produced, number of products assembled, and the like. After the matching has been accomplished, the informational input variable of a given agent represents the value of the informational output variable of the agent to which it has been connected.

Any structure developed during the integration process may be enclosed into an autonomous subsystem using ports. The informational input and output ports provide an interface to the subsystem environment. This interface allows one to develop hierarchical structures suitable for modelling of hierarchical decision making in manufacturing systems (Sethi and Zhang [8]).

4. GENERATION OF CONNECTIONS

It is easy to notice that without any additional mechanism for generation of connections, the bulk of the resulting matchings would not posses either a physical or logical meaning, and could lead to systems that might not be valid, for example, systems without a boundary. In addition, in the absence of constraints guiding the generation of connections, the number of all possible matchings for a complex systems (with large initial number of agents) would become difficult to manage.

Generation of connections between agents and/or subsystems is guided by production rules. The rules refer to all possible types of connections, i.e.:

(i) agent -----> agent,
(ii) subsystem -----> subsystem,
(iii) agent -----> subsystem,
(iv) subsystem -----> agent.

Production rules are structured to ensure that (Hassan et.al. [9]):

(i) an overall system is integrated in such a way that all agents are included. The set of agents at the initial state of integration represents all functions specified at the functional decomposition stage.
(ii) all boundary informational inputs and outputs of the system are included in the integrated system and they represent the only way that the system communicates with its external environment,
(iii) only physically and logically feasible matchings of agents can be explored (i.e., only feasible variants of the integrated system).

The agents and subsystems in the production rules supporting the integration are referred to as elements. An informational input boundary element is the one that accepts information from an environment of the system, and an informational output boundary element is the one that provides informational output to the environment. Two production rules that indicate whether the integration solution is feasible are presented next.

Rule 1

IF	there is only one element left
THEN	do not generate connections

Rule 2

IF	a single element that is left includes boundary inputs and outputs only
THEN	it is an overall system

Production rule 3 sets a starting point for generation of connections and thus the beginning of the process of overall system building. It requires that the integration process begins with an input boundary element.

Rule 3

IF	there are more than one element
THEN	select a connection for an input boundary element

Production rule 4 is used for inclusion of all elements in the integrated solution.

Rule 4

IF there are elements other than the boundary elements
THEN do not specify any connections that involve boundary elements only

Production rules 5 and 6 make sure that agents that have been defined as boundary elements at the representation stage create also the boundary of the overall system.

Rule 5

IF an element is an input boundary element
THEN it can not accept an input from any other element

Rule 6

IF an element is an output boundary element
THEN it can not provide an input to any other element

Production rules 7 and 8 generate the actual connections. Rule 7 sets the matching priority for agents with exactly the same informational input and output. Since it is not always the case, production rule 8 specifies the priority for generating the connections between elements with only partially identical input and output variables.

Rule 7

IF two elements have identical output and input variables
AND there are no production rules that prevent from connecting them
THEN specify the connections for these elements

Rule 8

IF there are no elements with identical input and output variables
AND there are elements with partially identical input and output variables
AND there are no production rules that prevent from connecting them
THEN specify the connection for these elements beginning with the closest match

The notion of the closest match can be explained with the following example. Assume that X1, X2, and X3 are the informational output variables of agent A1, and that there are no other elements with identical input variables. There are, however, elements A2 and A3 with input variables X1, X2 and variables X1, X4, and X5, respectively. The match between A1 and A2 (two identical variables) is closer than between A1 and A3 (only one variable is identical).

Production rule 9 develops a multilevel hierarchy of the inner structure of the overall integrated system.

Rule 9

IF a connection for an input boundary element has been specified
THEN continue with selecting connections for elements that have not been listed in the specifications

Production rule 10 is applied when all inner (not boundary) elements have been already included in existing subsystems but the integration process has not been completed.

Rule 10

IF there are boundary elements only
THEN specify connections between them

All the above production rules have been structured independently of the given domain and can not be modified by a system analyst. The analyst may, however, add domain-based production rules. They may follow, for example, the safety requirements, marketing requirements, or other constraints imposed by the analyst.

5. THE INTEGRATION ALGORITHM

Integration of agents and subsystems into an overall system begins at the lowest level (level 1) of abstraction represented by the initial number of agents generated at the agent representation stage. It is performed according to the algorithm presented next.

Integration Algorithm

Step 1. Open a set AGENT_BASE consisting of all agents generated at the representation stage.
Set level = 1.

Step 2. Apply production rules to generate connections between elements in AGENT_BASE.

Step 3. If no connections are generated, stop;
Otherwise, match elements in AGENT_BASE into pairs using the existing connections.

Step 4. Define informational input and output variables for subsystems generated by the matching process.

Step 5. Remove from AGENT_BASE all elements that have taken part in the matching process.

Step 6. Add to AGENT_BASE all subsystems generated by the matching process.

Step 7. Set level = level + 1 and go to Step 2.

According to the above algorithm, in order to enter the next level of integration it is enough to generate only one connection in Step 2. Elements of the agent base that are not matched at level i are considered for matching at level i+1. The algorithm terminates at the level at which it is not possible to match agents and/or subsystems into pairs (no connections are generated). Two cases are possible. First, matching is no longer possible because all agents and subsystems have been exhausted and an overall system has been built. Second, although there are subsystems and/or agents available, it is not possible to match them. In this case, the integration path explored has not led to the overall system and none of the subsystems at the highest level includes all functions specified in the functional space.

6. CONCLUSION

The design process of a synthetised hierarchical system is based on achieving a consensus among its various activities. The major activities of this process are functional decomposition, agent representation, and information flow integration. The outcome of the integration process is the overall system. In this paper, a rule-based approach was proposed for integration problem formulated as follows:

> Given the informational inputs and outputs of agents, find the overall system being designed that meets the desired functions and is integrated through the flow of information.

Elements of an agent base are integrated using an algorithm into an overall system that has a hierarchical structure. General production rules supporting generation of connections for agents and models were developed. The general production rules relate to the underlying systems theory.

They are structured independently of the design domain and can not be modified by a system analyst. Production rules ensure that only feasible variants of the developed system are explored.

7. REFERENCES

1. Biemans, F.P.M., Manufacturing planning and control, Amsterdam: Elsevier,1990.

2. Burch, J. and Grudnitski, G., Information systems theory and practice, New York: Wiley, 1989.

3. Szczerbicki, E., "Information flow evaluation in autonomous groups functioning", ", IEEE Transactions on Systems, Man, and Cybernetics, vol. 21, pp. 402-408, 1991.

4. Szczerbicki, E., "Acquisition of knowledge for autonomous cooperating agents", IEEE Transactions on Systems, Man, and Cybernetics, vol. 23, pp. 1302-1315, 1993.

5. Szczerbicki, E., "Autonomous manufacturing agent in static environment: qualitative reasoning for knowledge retrieval", Systems Analysis, Modelling, Simulation, vol. 22, pp. 75-84, 1996.

6. Bobrow D.G., "Dimensions of interaction", AI Magazine, Fall, pp. 64-74, 1991.

7. Albus, J.S., "Outline for a theory of intelligence", IEEE Transactions on Systems, Man, and Cybernetics, vol. 21 pp. 473-509, 1991.

8. Sethi, S.P. and Zhang, Q., Hierarchical Decision Making in Stochastic Manufacturing Systems, Boston: Birkhauser, 1994.

9. Haasan, A.A., Hershey, J.E., Schroeder, J., Sohie, G.R.L. and Yarlagadda, R.K.R., The Elements of System Design, New York: Academic Press, 1994.

INTELLIGENT DECISION SUPPORT SYSTEM FOR EQUIPMENT DIAGNOSIS AND MAINTENANCE MANAGEMENT

Yiliu TU, Jun ZHANG*
City University of Hong Kong
Tat Chee Avenue, Kowloon, Hong Kong

Eddie H.H. Yeung
Central Textiles (H.K.) Ltd.8-14 Sha Tsui Road, Tsuen Wand, N. T., Hong Kong

*To whom all correspondence should be addressed "Dept. of E.E., City Univ. of Hong Kong"
E-mail: 95411806@plink.cityu.edu.hk , Phone: 852-2788 9951, Fax: 852-2788 9135.
Who is responsible for the preparation of the final version of the paper.

ABSTRACT

In this paper, an AI (artificial intelligence) approach of the integrated maintenance management system is reported. The research work has been done on two levels. On the top level, which is named ***managerial decision support level.*** On the second level, which is named ***diagnosis and maintenance level.*** A number of diagnosis and maintenance Intelligent Decision Support Systems (IDSS) are developed based on the Bayesian theory or causal probabilistic networks (CPNs). In this paper, a generic CPN for the maintenance of open-end spinning mills is reported.

KEYWORDS:

Artificial Intelligence, Causal Probability Network, Intelligent Decision Support System, Maintenance Management And Control.

1. INTRODUCTION

Hong Kong manufacturing industry has always been exported-oriented. Its success has been attributed to its flexibility and adaptability to frequent changes of market demands and its versatility and industrious human resources. Nowadays, there are threats to its competitive position. These include the advanced technological pressure from the developed nations, competitions from other South East Asian countries. To enhance their competitive positions, many manufacturing companies in Hong Kong are moving towards to so-called World Class Manufacturing (WCM) [3]. This transformation leads a number of new problems for the manufacturing companies in Hong Kong. Central Textile (HK) Ltd., for example, to compete with its competitors is now moving its outdated production facilities to the mainland China to keep its traditional products production in China and at meanwhile it also invests on a number of new facilities in its factory in Hong Kong to produce new products. In its factory in Hong Kong, in addition to the investment on the equipment, the company also adopt the new managerial philosophy, such as Just in Time production, lean production [4], total quality management and world class manufacturing. These changes require the decisions on the maintenance activities, besides of the equipment working conditions, have to be made also under an integrated or comprehensive consideration of cost, quality and production efficiency. In fact, the problems in Central Textile (HK) Ltd. As mentioned above are common problems for the other manufacturing companies in Hong Kong. This research project aims to tackle these problems.

This paper reports the research project which has been carried out at the Central Textiles (H.K.) Ltd. in Hong Kong. The main objective of the project is to help the company to approach an integrated maintenance management system by using Artificial Intelligence (AI) technologies. This Al-based maintenance management system schedules and controls maintenance activities under considerations of not only the working conditions of equipment in the company but also the production quality, production system efficiency and production cost.

The research work has been carried out on two levels. On the top level which is named ***managerial decision support level.*** The overall maintenance management system is designed by

GRAI method. On the second level which is named ***diagnosis and maintenance level.*** A number of diagnosis and maintenance Intelligent Decision Support Systems (IDSS) are developed based on the causal probabilistic networks (or Bayesian networks). These IDSSs support decision making on maintenance activities planning and control under consideration of equipment working conditions, maintenance cost, product quality and a particular machine working efficiency. An IDSS for the open-end spinning mills is developed as a generic model and reported in this paper. According to the basic principles of this generic module, the IDSSs for other type of machines in the company can be further developed. Those IDSSs are the core modules of the decision centers in the overall maintenance management system which modeled by GRAI method on the top level or managerial level. By this new Al-based integrated maintenance management system, the company, rather than its traditional "run-fail-fix" maintenance management strategy, is approaching to a more dynamic and integrated maintenance management strategy to enhance its moving towards to WCM.

2. BRIEF INTRODUCTION OF HUGIN EXPERT SYSTEM

To support the development of this type of IDSS, a software shell which is named HUGIN was developed in Aalborg University, Denmark in 1993 [28, 29]. In the HUGIN system, a CPN is expressed by a number of nodes and probabilistic links. The factors which influence the inference problems and results are nodes in the CPN and the relations between the nodes are represented by the conditional probabilities which are derived from the experience data and calculated according to the Bayesian rules.

$$p(s_j|o_j) = \frac{P(o_j|s_j)P(s_i)}{\sum_{k=1}^{k=n} P(o_j|s_k)P(s_k)}$$

The probabilities on the right hand side of above equation are called experimental probabilities. These experimental probabilities are predetermined according to the historical data records.

3. GENERIC IDSS FOR OPEN-END SPINNING MILLS

To support the decision making in the decision center "maintenance synthesis", a number of IDSSs are developed based on the similar system structure and principles. In this section, through the discussions of the IDSS for open-end spinning mills, it aims to illustrate this generic control structure and the principles.

3.1 The CPN of Maintenance Decision Model:

On the top level of the IDSS for any type of equipment, there is a maintenance decision model. This decision model is designed as a four-nodes CPN. These four nodes are respectively named M-Decision (maintenance decisions),Quality, Efficiency and Cost. The node M_Decision has three states, viz. *Corrective, Condition, Chance.* These three states are the possible outputs of the IDSS. The state *Corrective* means corrective maintenance. The state *Condition* means conditional maintenance. The state *Chance* means chance maintenance.

Except the node M_Decision. the other three nodes can be further extended into three CPNs which are named Quality model, Efficiency model and Cost model. Among these three model, the Efficiency model is a diagnostic model of the equipment. This is because a failure or a potential failure in a machine normally results in decreasing the machine efficiency. To the different types of equipment, the states in each of these three nodes may be different, and the CPNs of these three nodes also may be different. However, the design principles are almost same. The CPN of Maintenance Decision Model for open-end spinning mills is shown in Figure 1. In the following sub-sections, the Quality, Efficiency and Cost models for the open-end spinning mills as shown in Figure 1 will be discussed. As an example, the Quality model will be discussed in the following sub-sections aim to give readers an overview of general ideas on how to design a diagnosis and maintenance management IDSS for any type of equipment in the Central Textile (H.K.) Ltd.

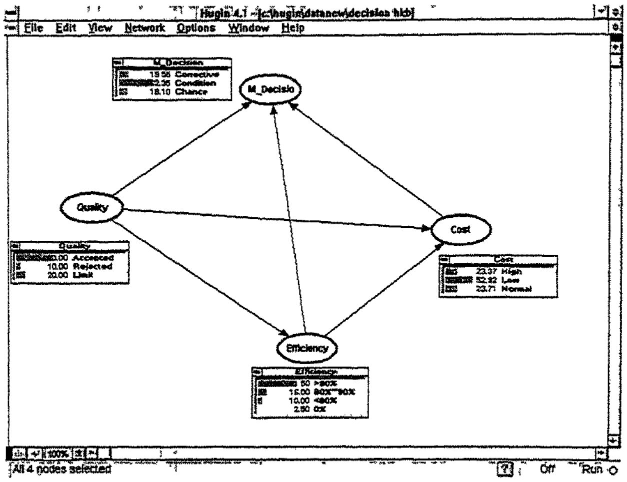


Figure 1 The CPN of maintenance decision model for open-end spinning mills

3.2 The Quality Model of Open-end Spinning Mills ·

In the Central Textile (H.K.) Ltd., due to the following reasons, 'un-necessary' maintenance actions may be carried out.

· The maintenance technicians have no experience in judging whether the quality is accepted or rejected according to different quality requirement

· Some of the technicians are afraid of the responsibility of making bad yarns. Hence they repair and/or replace the parts just 'for sure'. Therefore, the quality factor has to be taken into account in the maintenance decision making. At the moment, the company's quality policies for the yarn produced by open-end spinning mills are briefly summarized below:

1) To monitor the yarn quality of the open-end spinning machine by the on-line quality system and to eliminate all the yarn faults which are not accepted by the machine itself.

2) To check the quality of the open-end yarns by the testing equipment in the laboratory weekly or on some occasions such as customers' complaints, equipment faults happened on random sampling methods. This is to make sure that the on-line quality system is working on the condition of problem free and to ensure the quality of the products.

3) To check the yarn quality with the subject to the following parameters:

According to the company's quality policies, the quality model for open-end spinning mills is developed as shown in Figure 2. It is a CPN which consists of 11 nodes. Some of these 11 nodes are briefly explained in the following.

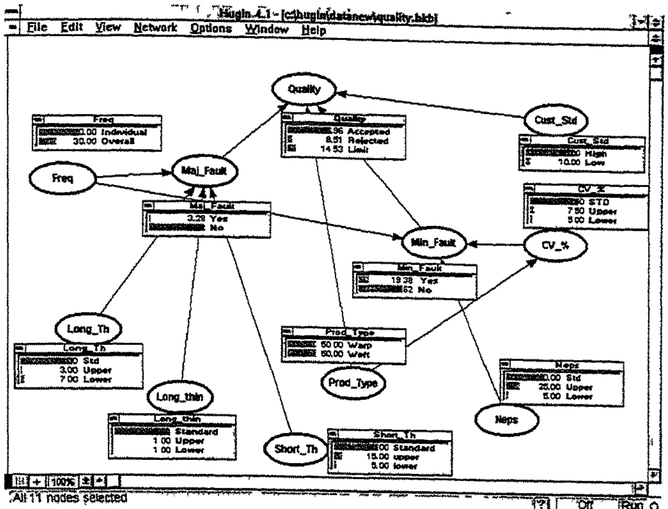


Figure 2. The quality model for the open-end spinning mills

Quality: It has three states which refers to the results from the quality control (QC) department, i.e. *accepted, rejected, limit. limit* means the product does not satisfy the quality requirements, but it may satisfy with the requirements of some customers.
Cust_Std: It refers to customer standard. It includes two states, viz. *high* and low. *high* means the customers really concern with the product quality and the product needs to be tightly controlled under the quality policies. low means the customers do not mind the quality of their product very much. Hence the product can be out of the company's quality standard with in a limited domain
Prod_Type: It refers to product types. Hence it includes two states, warp and weft.
Mai_Fault: It refers to the major quality problems, i.e. long thick places (as node long_Th shown in Figure 2), long thin places (as node Long_Thin shown in Figure 2), short thick places (as node Short_Th shown in Figure 2). If any one of these quality parameters out of the standard, the state of Mai_Fault will be *high.* Otherwise it will be low.

The state probabilities (or basic probabilities) in the CPN of the quality model as shown in Figure 2 were derived according the data collected from experts in the QC department. According to a particular case (let us say that there is a minor quality problem of neps and the product type is warp), the CPN as shown in Figure 2 will give an influence upon or support to the maintenance decision making.

4. CONCLUSIONS AND FUTURE DEVELOPMENTS

Through the description made in this paper, the following statements can be concluded.

1) A prototype of an integrated maintenance management system has been developed for the Central Textile (H.K.) Ltd. and proposed in this paper. This integrated maintenance management system aims to provide supporting decisions in the maintenance management under wider considerations of the machine conditions, production efficiency, maintenance cost, and product quality.

2) An intelligent decision support system (IDSS) for fault diagnosis and decision making support on maintaining the opened spinning mills has been developed based on the Bayesian network or CPN. In this way of approach of IDSS, the inference procedure of human beings can be better emulated. The knowledge and experience

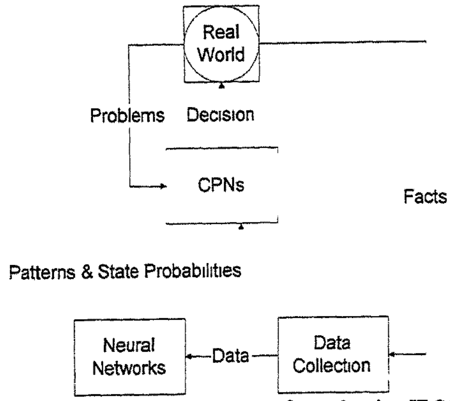


Figure 3 A framework of an adaptive IDSS

can be simply expressed by the conditional probabilities and the system gains higher intelligence. The developed models are expected to be used in the company in the near future.

3) Since the design of the system is not limited by a particular control objective,

Figure 3 A framework of an adaptive IDSS. The IDSS presented in this paper can be considered as a generic model for the design of other models in the integrated maintenance management system.

4) The diagnostic CPN reported in this paper is an effective and feasible diagnostic system. It can help maintenance technicians to quickly allocate the causes (or sources) of problems. It can be also used as a training tool for improving knowledge and skills of maintenance workers.

In order to improve the diagnostic accuracy and adaptive ability to the changes in the environment (e.g. new machines or production system), an "adaptive" or self-leaning IDSS is needed. This type of adaptive IDSS shall be able to automatically update its inference algorithms (e.g. architecture of the CPNs) or parameters (e.g. state probabilities in the CPNs) through self-leaning from the real world. A framework for an adaptive IDSS is proposed in Figure 3. As shown in Figure 3, the inference engine of the IDSS is developed by CPNs and the neural networks are used for the pattern identification and state probability assessment. This type of IDSS is expected to make more accurate decisions as long as it has been implemented.

REFERENCES

1. EUREKA, European Benchmark Study on Maintenance, EBSOM-EU 724, 1993.
2. T. Wireman, *World Class Maintenance Management,* New York: Industrial Press, 1990.
3. R. Discenza, *Creative Process to Develop World Class Svstems forManufacturing,* University of Colorado, 1990.
4. P.M. Gallois, NThe new industrial challenge: Towards a 'Lean Production'", *Advances in Production Management Svstems, IFIP Transactions* B-13, pp. 3-10, 1993.
5. P.T. Kidd, *Agi/e Manufacturing: Forging /IJew Frontiers,* Addison-Wesley, 1994.
6. M. Klein, and L.B. Methlie, *Expert Svstems: A Decision Support Approach,* Addison Wesley Publishing Company, New York, 1990.
7. M. Santana, "Managerial Learning: A Neglected Dimension in Decision Support Systems," Proceedings of the 28th Annual Hawaii International Conference on System Sciences, Vol. IV, pp. 82-91, 1995
8. A. Borden, "Designing and Maintaining Decision-making Processes," AGARD Conference Proceedings 545, Aerospace Software Engineering for Advanced Systems Architectures, Paris, France, pp. 311-6, 10-13 May 1993.

9. M.T. Liberatore and A.C. Stylianou, "Using Knowledge-based Systems for Strategic Market Assessment," Information and Management, Vol. 27, Iss. 4, pp. 221-32, October 1994.
10. T.L. Saaty, *Multicriteria Decision Making, The Analvtic Nierarchv Process,* RWS Publications, 4922 Ellsworth Avenue, Pittsburgh, PA 15213, USA, 1990
11. M.A.P. Davies, M.A.P, "A Multicriteria Decision Model Application for Managing Group Decisions," Journal of the Operational Research Society, Vol. 45, Issue 1, pp. 47-58, 1994.
12. C. Kocourek, "A Petri Net Based Design Decision Support System," Proceedings of the IASTED International Conference, Applied Modelling and Simulation, Vancouver, B.C, Canada, pp. 108-114, 21-23 July 1993,
13. A.R. Hurson, S. Pakzad, and B. Lin, B, "Automated Knowledge Aquisition in a Neural Network Based Decision Support System for Incomplete Database Systems," Microcomputers in Civil Engineering, Vol. 9, Issue 2119941, pp.129-143, 1994.
14. T.S. Quah, C.L. Tan, and B. Srinivasan, "A Fuzzy Neural-network Decision Support Systems for Bond Trading, Proceedings of International Symposium on Artificial Intelligence," The Artificial Intelligence Technology Transfer Conference, Monterrey, Mexico, 20-24 September 1993.
15. A. Ishikawa, "The New Fuzzy Delphi Methods: Economization of GDS IGrouP Decision Supportl," Proceedings of the Twenty-sixth Hawaii International Conference on System Socience, Wailea, HI, USA, Vol.4, pp. 255-264, 5-8 January 1993.
16. C.P. Schrunder, J.E. Gallertly, and J.R. Bicheno, "A Fuzzy, Knowledge-based Decision Support Tool for Production Operations Management,n Expert Systems, Vol. 11, Issue 1, pp. 3-11, February 1994.
17. P.G.W. Keen, "Value Analysis: Justifying Decision Support Systems," Management Information Systems Quarterly, 1980.
18. E. Charniak, "Bayesian Networks without Tears", Al Magazine, No. 12 pp. 50-64, 1991.
19. S. Holtzman, *IntelligentDecision Svstems,* Addison-Wesley Publishing Company, Reading, MA, 1989
20. H.A. Simon, *The llew Science of Management Decisions,* Harper & Row, New York, 1960.
21 A. Newell and H.A. Simon, *Iluman Problem Solving,* Prentice-Hall, Englewood-Cliffs, NJ, 1972.
22. H. Mintzberg, D. Rasinghani, and A. Theoret, "The Structure of the Unstructured Decision Processes," Administrative Science Quarterly, Vol. 21, pp. 246-275, 1976.
23. Y.L. Tu and E.H.H. Yeung, "Intelligent Decision Support System IIDSS) for Equipment Diagnosis and Maintenance Management," Proceedings of 3rd International Conference on Manufacturing Technology in Hong Kong, 1995, pp. 658-667, December 13-16.
24. Y.L. Tu, H.C.S. Leung, and E.H.H. Yeung, "Intelligent decision support system for open-end spinning mills diagnosis,n Proceedings. of Vll Workshop on Supervising and Diagnostics of Machining Systems, Karpacz, Poland, 11 -15 March 1996.
25. J.T. Luxhoj, "An Organizational Scheme for Maintenance Modeling, Logistics Spectrum," Vol. 25, No. 1, pp. 25-33, 1991.
26. J.T. Luxhoj, J.T. Charles and M. Rao, "An Intelligent Maintenance Support System IIMSS) Concept, Proceedings of the 26th Symposium of the Society of Logistics Engineers,n Fort Worth, TX, August, pp. 117-126.
27. G. Doumeingts, E. Dumora, M. Chabanas, and J.F. Huet, "Use of GRAI method for the design of an advanced manufacturing,D Proceedings. of 6th International Conference on Flexible Manufacturing Systems, pp. 341-358, November 1987.

APPLICATION OF NEURAL-NETWORK FOR EFFECTING A KNOWLEDGEABLE MANUFACTURING SYSTEM IN REAMING

P.G.Mathews and M.S.Shunmugam
Manufacturing Engineering Section
Department of Mechanical Engineering
Indian Institute of Technology,
Madras-36, India.
(Email : mech7@iitm.ernet.in)

ABSTRACT

In this paper, a new approach is presented for accomplishing a knowledgeable manufacturing system enabling status identification in reaming using Artificial Neural Network (ANN). The force, vibration and acoustic emission parameters were obtained during reaming process and used as inputs for the ANN.The network was able to predict surface finish and roundness error more effectively with sensor fusion.

KEY-WORDS

Hole quality, Reaming, Knowledgeable Manufacturing system, Artificial Neural Networks, Sensor Fusion.

1. INTRODUCTION

Functional behaviour of components depends on surface quality. In modern precision assembly requirements, hole quality is very important. These requirements can not be generally met by conventional twist drills, though these end-cutting tools produce holes in solid material economically. In practice, holes are machined with twist drills and reaming is carried out as a second operation. In the interchangeable system of manufacture inprocess measurement and control of quality is very important.

In-process and on-line measurement of surface finish on external surfaces through optical techniques have been reported by Russell[1] and Kimiyuki[2]. Similarly a laser based in-process cylindricity measurement in boring operation is reported by Kim[3].But in the case of reaming since the tool is long and having full contact it is very difficult to have on line direct measurement of surface finish.and form error.Therefore there is a need for indirect methods for the prediction of surface finish and form error in the case of reaming.

2. INTELLIGENT PROCESS CONTROL

Many sensing techniques have been proposed for metal cutting process monitoring.For the process performance, indicators such as cutting force, vibration ,temperature, acoustic emission etc. are acquired using intelligent sensors as reported by researchers Li[4] and Tlusty[5].These are indirect techniques sensitive to process changes. Hence monitoring these parameters is the primary function of any knowledgeable manufacturing system. For the spontaneous decision making on the performance earlier researchers were concentrating on acquiring a single process signal to decide on machining status. However information from only one sensor is generally insufficient for decision making because of the stochastic nature of the machining process; hence integration of data from several sensors operating simultaneously is the solution.

2.1 Acoustic Emission and Other Sensor Data

Out of all the process performance indicators the acoustic emission methodology is attractive because the signal (ie the transient elastic stress wave) is generated by the major activities of metal cutting, including plastic deformation, frictional (rubbing) contact and fracture of both chips and tool.Because of this close correlation with metal cutting it is the most promising indirect monitoring technique. Along with acoustic emmision, vibration and force signals are also used.

2.2 Sensor Fusion

Sensor fusion is the process of integration and co-processing of data obtained from different sensor types inorder to increase the performance of the process monitoring .Erdal Emel et al.[6] used force and acoustic emission signals for improving the progressive wear detection performance as well as tool breakage monitoring..Thus researchers have begun to look at ways to collect the maximum amount of information about the state of a process from number of different sensors. David A.D [7] and Jeong-Du Kim [8] have used multisensor approach for condition monitoring . The objective of sensor fusion is to improve the reliability by integrating the information from variety of sensors with the expectation that this will increase accuracy in the knowledge about the environment.

3. EXPERIMENTAL DETAILS

The machining trials were carried out on a horizontal boring and milling machine(SCHARAMANN-WB75), as speed and feed ranges necessary for the present investigations were available only on this machine..EN4 steel work. samples were used. The reaming was done using a LH helical flute HSS reamer 20mm diameter (IS 5445-1978). Cutting parameters are given in Table.1. Soluble oil was used as cutting fluid in drilling as well as reaming operations. Experimental setup is shown in Fig.1.

Kistler 9271A drilling dynamometer is used to measure thrust and torque. Brüel & Kjær Vibration analyzer Type 2515 is used to measure acceleration .Acoustic emission was sensed using a piezoelectric sensor ,preamplified (60db) and then processed through the AET 5500 system. The acoustic emission system parameters such as ring down count, rise time, event, event duration and energy were used for analysis. Surface roughness of the reamed holes were measured using Perthometer and roundness error using Perthen formtester..

Table 1 Machining Parameters

Cutting Speed (v m/min)	6,8,10,12
Feed (s mm/rev)	0.18,0.3,0.48,0.62
Reaming allowance (mm)	0.25,0.5

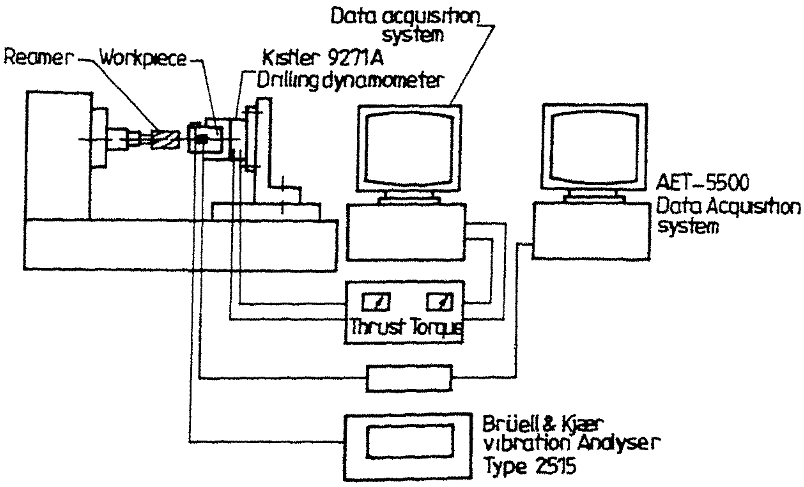


Fig. 1 Experimental setup

4. RESULTS AND DISCUSSION

4.1 Surface Roughness and Roundness Error

As cutting speed, feed or reaming allowance increases surface roughness increases.This agrees with the results published earlier by PERA [9] and Tilsley[7]. Roundness error also seems to be increasing with cutting parameters.This may be due to increase in vibration of the reamer with increase in cutting parameters.

4.2 Thrust and Torque

In general thrust and torque show increasing trend with increase in the cutting parameters. But due to the random variations in machining process there is much fluctuations in the thrust and torque signals.

4.3 Acoustic Emission Parameters.

Typical observations of AE characteristics with changes in cutting speed at 0.48mm per rev. feed and 0.5mm reaming allowance is shown in Fig.2 Initially as the cutting speed increases AE parameters show an increasing trend associated with normal material deformation and then it decreases .The decreasing trend can be attributed to burst type signals.This may be due to chip clogging as speed increases , which in turn can scratches the machined surface. Similarly as speed increases vibration increases which can lead to unsteady cutting.

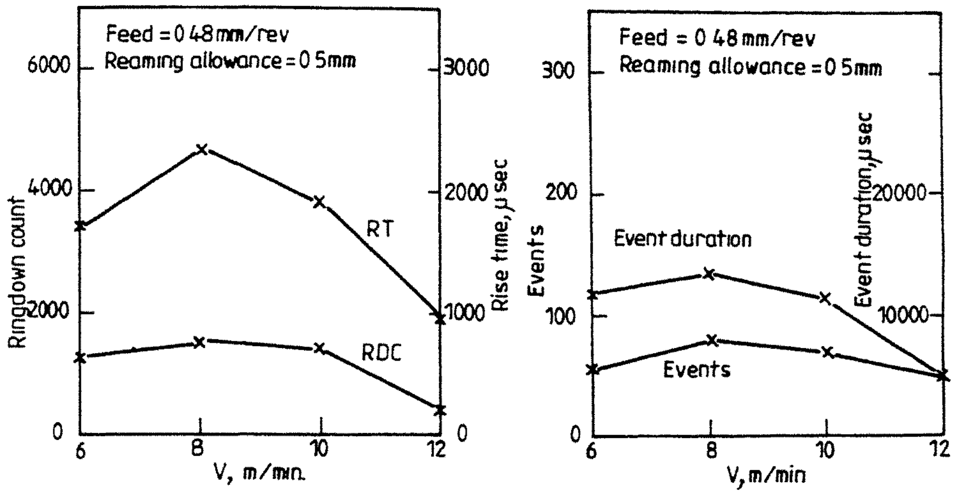


Fig.2 Variation of AE parameters with cutting speed

4.4 Vibration.

Fig.3 shows the acceleration values and surface finish values corresponding to cutting speed variation.Because of the over hang of the reamer vibration amplitude increases as speed or feed increases.This results in an increase in roughness.

4.5 The Performance of Multi layer Neural Network.

A multi layer neural network was trained using the data obtained, with the highly popular Back Propagation Algorithm(BPA) for weight updation. The architecture of multi layer neural network is shown in Fig.4. The main advantage of an ANN is its ability to learn from the environment and to improve its performance through learning.ANN learns about its environment through an iteration process. Each iteration consists of two passes: forward and backward. The signals from the sensors are given to the input nodes and its effects passes layer by layer to the output nodes. In the backward pass, the error between the desired and the actual output is used to adapt the synaptic weights to store the knowledge.The network becomes more knowledgeable about its environment with each iteration.

Performance of the neural network with cutting force, vibration and A.E parameters is shown in Fig.5 .It shows the efficiency of ANN in predicting the surface roughness and roundness error in reaming.

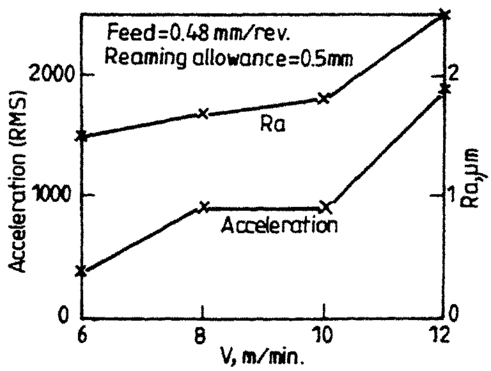


Fig.3 Variation of vibration parameter with cutting speed

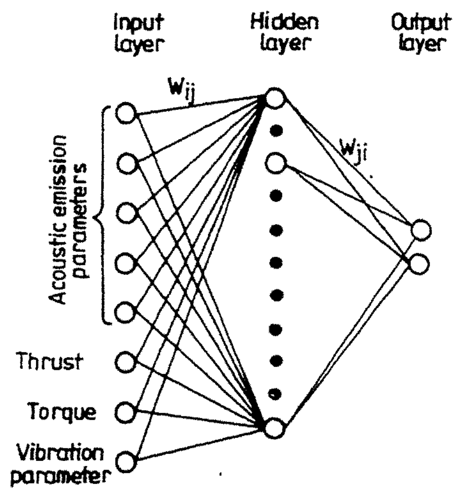


Fig. 4 Architecture of multi layer Neural Network

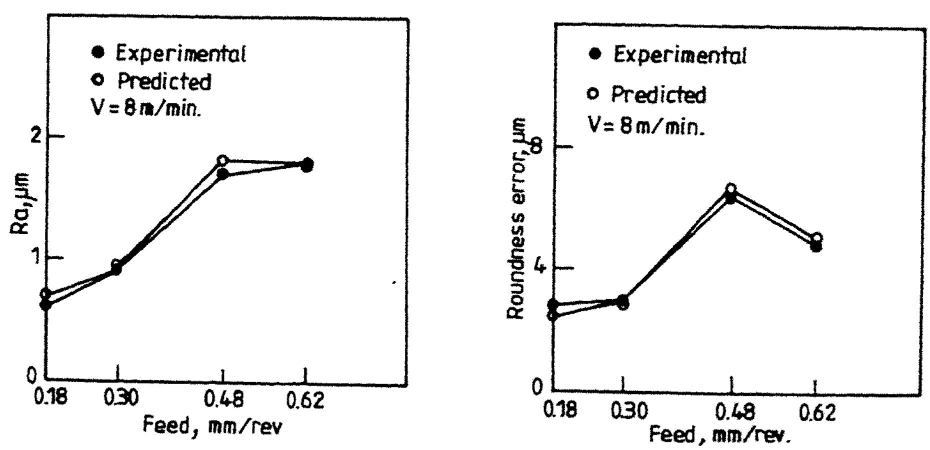


Fig.5 Comparison of Neural Network output with experimental results

5. CONCLUSIONS

Random variations in the machining process can affect the force, vibration and AE signals. So depending on only one parameter for analysis is not sufficient. Force, acoustic emission parameters and vibration parameter as inputs and surface finish (Ra value) and form error (roundness error) as output, the network gives much superior results, clearly indicating the more effective nature of ANN in indicating the machining status. Hence sensor fusion approach through ANN is the most effective methodology for implementing the knowledgeable manufacturing system in reaming.

6. REFERENCES

1 Russell, D.Y etal., "In-process and on-line measurement of surface finish", Annals of CIRP 29/1, pp. 435-440,1980.

2 Kimiyuki Mitsui & Hisayoshi Sato., "Development of an in-process sensor for surface roughness by laser beam", 21st MTDR conf.,1980.

3. K.Kim & etal., "In-process control of cylindricity in Boring operations", Jl. of Engg. for Industry vol.109, pp. 291-296, Nov 1987.

4. Li Dan and J Mathew, "Tool wear and failure monitoring techniques for turning- a review", Int.J.M.T. Manuf., vol.30,No.4, pp. 579-598,1990.

5. Tlusty.J and Andrews,G.C "A critical review of sensors for unmanned machining.", Annals of CIRP 32/2, pp. 563-572 ,1983.

6 E.Emel etal "Acoustic emission and force sensor fusion for monitoring the cutting process." Int.J.Mech Sci., vol.31, pp. 795-809, 1989.

7 David A.D "Neural Network sensor fusion for Tool Condition monitoring," Annals of the CIRP., vol. 39/1, pp. 101-105,1990.

8 Jeong-Du Kim & In-Hyu choi, "Development of a tool failure detection system using multi sensors." Int.J.Mach.Tools Manufact., vol. 36, No.8 , pp. 861-870 ,1996.

9. PERA Report ,"Effect of reaming conditions on hole accuracy and surface finish", 1956.

10. Tilsley.R., "Development and applications of PERA Reamer", Annals of CIRP., vol.15, pp. 137-145,1967.

OBJECT ORIENTED SIMULATION OF RELAY LADDER LOGIC

J. A. Ferreira, J. L. Azevedo, J. P. Estima de Oliveira
University of Aveiro
Portugal

ABSTRACT

Since their appearance, Programmable Logic Controllers (PLCs) have gain a very strong position in the industrial automation field. The growing complexity of the applications using this type of equipment strongly depends on programmers and maintenance personnel, and so more and better PLC education is of great demand nowadays

The present work deals with an object oriented simulator for the relay ladder logic language. The simulator is a module of a more general system that emulates a real PLC. This system provides a user friendly graphical interface and allows PLC programming and debugging. The simulator runs as a Microsoft Windows application and was developed using C++. With the same algorithm two simulation modes, allowing different analysis of PLC programs, were implemented: fast simulation and real time simulation.

KEYWORDS

Programmable Logic Controllers, Ladder Diagrams, Object Oriented Programming, Object Oriented Simulation.

1. INTRODUCTION

In the sixties, the need to reduce the costs of the frequent changes in the relay based industrial control systems lead to the development of the concepts associated with programmable logic controllers. The reliability and the flexibility, along with the simple programming language initially used, are some of the factors which influenced their strong implantation in the industrial environment. Regardless of the size, and complexity, almost all PLCs have the same basic components and the same functional characteristics [1].

The spreading of the PLC market and the growing complexity of the applications increased the need of programmers and maintenance personnel, as also the need of more and better training. Lower costs of formation in automation control programs can be achieved through the use of simulation tools. This idea was successfully implemented, namely in instructing industrial controls using ladder diagrams [2].

Our work deals with the design of a virtual PLC (VPLC), intended to be used as a (PLCless) tool to teach relay ladder programming and to develop and debug PLC applications [3]. The VPLC runs in a personal computer (PC) under the Microsoft Windows, and was built using object oriented techniques with C++ [4].

The basic architecture for a real PLC, and the proposed architecture for the virtual PLC are shown in figure 1. In a real PLC, a Central Processing Unit (CPU) executes cyclically a program contained in the program memory. For each cycle, the inputs are acquired at its beginning and, in accordance with them and with the control program, the outputs are updated at its end. During program execution, the CPU uses another PLC memory to store and transfer data.

In the virtual PLC the program is written (using the Relay Ladder Language) and supported by the programming editor. In run mode, the stimuli editor reads the inputs, the VPLC makes the simulation of the ladder program and modifies its memory in accordance with the inputs and the program; at the end it updates the outputs and show them, in a window, using the output graphic module.

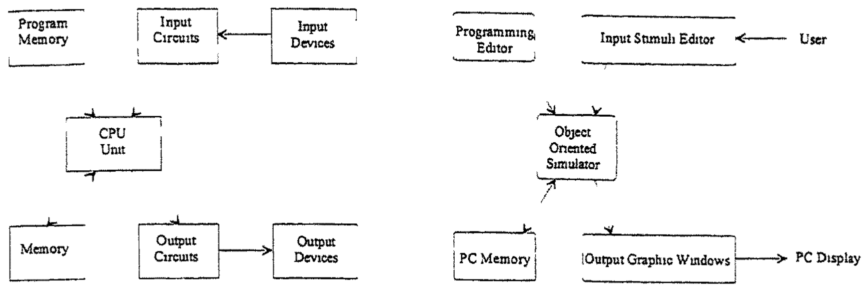


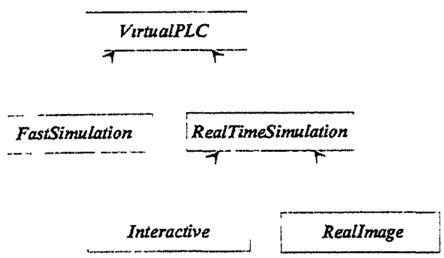
Fig. 1 - Real and Virtual PLC architectures

2. PRESENTATION OF THE SIMULATION MODES

Two simulation modes, allowing different analysis of the results of a program, has been implemented· fast and real time simulations. In the fast simulation mode, the program is executed without pauses and the results are shown in the same stimuli editor that was used to draw the inputs.

The operation of the real time simulation mode is similar to the fast, but the scan cycles are controlled by a clock. The results of this mode can be visualized in two different ways: in a window containing a picture of the real PLC or in an interactive window where the inputs can be modified on-line by the user.

An hierarchy of virtual PLCs were designed, in order to support different simulation modes. This way, the design of a new simulation mode is easy to obtain by deriving a new descendent class. Figure 2 presents the hierarchy for virtual PLCs and the most important data and methods of the class *VirtualPLC*.




```
class VirtualPLC {
protected
static LadderProgram *prog,
InputStimuli *inputStim,
OutputResults *outputRes,
public:
virtual BOOL VOID InitSimulation(VOID);
virtual BOOL LadderSimulation(VOID),
virtual VOID UpdateOutputStimuli(WORD oStepTime),
virtual VOID UpdateInputStimuli(WORD oStepTime),
virtual VOID StopSimulation(VOID){}
},
```

Fig. 2 - Virtual PLC class hierarchy and part of the base class *VirtualPLC*

The *LadderProgram* object supports the ladder program inside the PC memory. Basically it is a two dimension array of pointers to ladder objects, but also contains the rules of ladder programming.

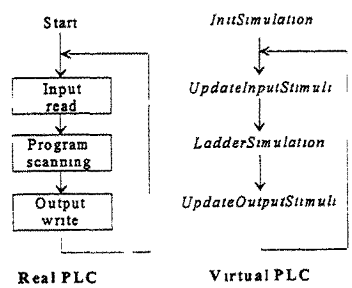


Fig. 3 - Real and Virtual PLCs in run mode

Being a cyclic machine, a PLC executes a program indefinitely, and each program execution is named as a scan. The virtual PLC simulates program execution in a similar way. Figure 3 represents the actions of a real PLC and of the virtual PLC during a scan. *InitSimulation*, *UpdateInputStimuli*, *LadderSimulation* and *UpdateOutputStimuli* are virtual methods defined in the base class *VirtualPLC*.

The virtual method *UpdateInputStimuli* reads the inputs defined in the stimuli editor and updates the respective inputs inside the virtual PLC. The virtual method *UpdateOutputStimuli* refreshes the output graphical windows, after each program scan. The virtual method *LadderSimulation* controls the simulation of a scan of the ladder program; it sends events to stimulate the objects that represent the ladder symbols (each object type contains its own simulation routine).

Fig. 4 presents a printout of a simple program to control a timed coffer. The order to open the coffer (output *Coffer*) only takes place after five clicks on the *Open* push button, followed by a 4 seconds wait period. To close the coffer the delay is 2 seconds after 3 clicks on the *Close* push button. The figure shows the ladder editor, with the ladder program; the stimuli editor, also showing the fast simulation results; the image of a real PLC with blinking lights; and an interactive window performing real time simulations.

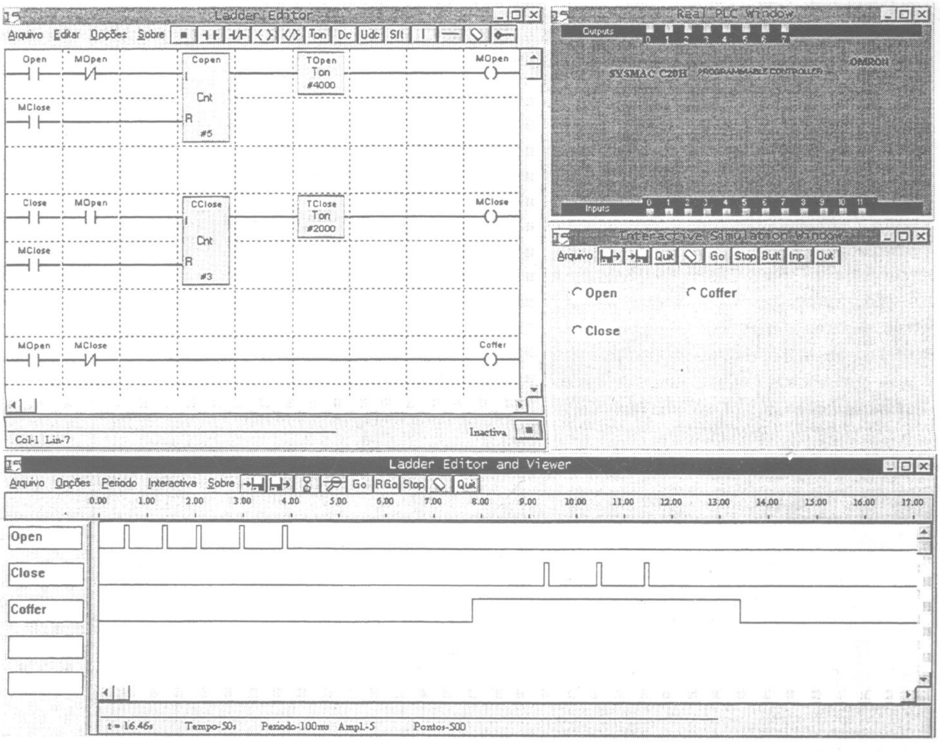


Fig. 4 - Layout of the virtual PLC system with the coffer example

3. RELAY LADDER SIMULATION ALGORITHMS

The advantages of the object oriented techniques in the implementation of simulation algorithms are already accepted. The design of a simulator in such a way involves the communication between objects along the time [5]. In the present case, the algorithm is based on an information passing scheme, using simulation events issued by objects lying in the left side of the ladder diagram and received by the objects located to the right. Simultaneously, the contents of the relevant memory positions are modified in accordance with the type and the action of the objects, which are interpreted in the context of the ladder program.

In the following example (figure 5), stating the logic equation $a.\overline{b} = c$, the execution of the instruction labelled **c (output)**, depends on the logic continuity from node 0 (the start of continuity) up to node **2**. This signal corresponds to the execution condition of the instruction **c**. But before this condition has arrived to node **2,** it must reach node **1**. Thus, the token results from the auto-simulation of the most-left ladder object. The execution condition of one instruction will be false if there is no logic continuity up to the left node of the object, and it will be true if there is logic continuity.

Another question to consider takes place when one or more objects are connected vertically. Considering the example of the figure 6, the logic continuity at node **(1,1)** depends on the objects **a, b** and **c**. This suggest that the nodes **(1,1)**, **(1,2)** and **(1,3)** are the same. In fact, the execution condition of **d** can be achieved by the logic continuity through any of the objects **a, b** or **c.**

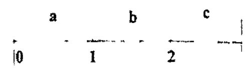


Fig. 5 - Ladder diagram representing the logic equation $a.\overline{b} = c$

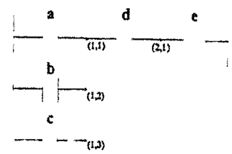
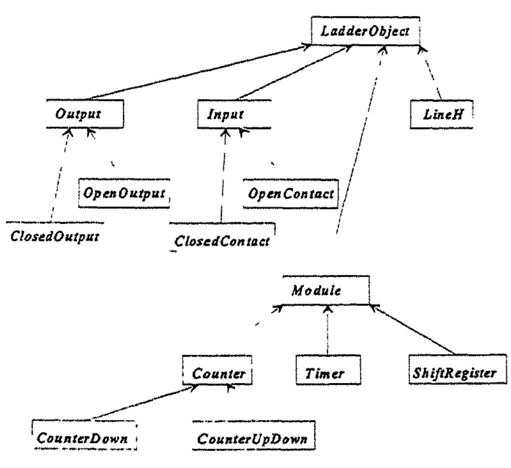


Fig. 6 - Ladder diagram representing the logic equation $(a + b + \overline{c}).d = e$

All the ladder symbols allowed by the ladder editor are objects (instances) of predefined C++ classes. These classes belong to a hierarchy of classes illustrated in figure 7. The base class *LadderObject* defines the functionality that descending classes should have. For instance, each ladder object has a name, a location in the programming editor, and the information about the connections to other symbols, as well as methods that are called to perform operations such as drawing (to draw itself) and simulation (invoked when the symbol behaviour is to be simulated).

The simulation of a ladder program is supported by the virtual methods *Simul* and *OwnerSimulation*, that are defined in the base class ladder objects, *LadderObject*. However, they can be redefined in other classes. The connection between the graphical symbols (and the corresponding ladder objects that represent those symbols) define the way as the information is exchanged between them.




```
class LadderObject {
protected:
   CHAR *objectName;
   CHAR **objectLabel;
   WORD *memLocation;
   Position gridPosition;
   LadderObject **leftObjects;
   virtual VOID OwnerSimulation()=0;
public:
   virtual BOOL Simul(VOID);
   virtual VOID ResetValuesToFirstScan(VOID);
   virtual VOID ResetToOtherScan(VOID);
   virtual VOID Draw(VOID);
   virtual VOID UnDraw(VOID);
};
```

Fig. 7 - Hierarchy of ladder classes and a part of the class LadderObject

For each ladder object, the virtual method *Simul* can be considered as the interpreter for the connections with other ladder objects, and the *OwnerSimulation* method executes the action of the instruction associated with the ladder object.

The goal of the method *LadderSimulation* is the stimulation of the simulation process; the stimulation is done by the invoking the *Simul* method for all terminal objects. This starts a navigation through the ladder editor grid, from the right to the left, followed by the return to the initial object, in a way similar to the simulation process implemented in [6]; this navigation takes advantage of the re-entering characteristics of the virtual method *Simul*. To simulate the action of one object, the input parameter (execution condition) should be known; this is obtained by sending a simulation event (calling the *Simul* method) to the left-connected objects. The right to the left navigation ends when the start of continuity is reached or when a ladder object, already simulated in the present scan, is found.

The left to the right navigation process results from the returning of the successive calls of *Simul* method from the different objects. For each ladder object, its *OwnerSimulation* method is also executed.

For all the ladder objects occupying one single cell (only one input), the interpretation of the program context is the same, thus the *Simul* method, defined in the *LadderObject* class, serves all the single cell objects.

Fig. 8 illustrates the sequence of the successive calls of the method *Simul* for the objects in a simple ladder diagram.

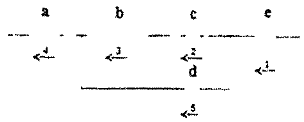


Fig. 8 - Sequence of the successive calls of the method *Simul*

For ladder objects with more than one input the *Simul* method should be redefined, because it needs to interpret adequately the signals connected to its inputs. As an example, the following list presents the differences between *Simul* methods for *LadderObject* and more complex *CounterUpDown* classes.

```
BOOL LadderObject :: Simul()
{
  if (flagSimulation)
    return (tokenSimulation);
  int i = 0;
  while (leftObjects[i++]) {
    tokenSimulation = tokenSimulation || leftObjects[i]->Simul();
    OwnerSimulation();
  }
  if( startContinuity ) {
      tokenSimulation = TRUE;
      OwnerSimulation();
    }
  return (tokenSimulation);
}
```

```
BOOL CounterUpDown :: Simul()
{
  if (flagSimulation)
    return (tokenSimulation),
  if (leftShape[0])
    counterUp = leftObjects[0]->Simul();
  if (leftShape[1])
    counterDown = leftObjects[1]->Simul();
  if (leftObjects[2])
    reset = leftObjects[2]->Simul();
  OwnerSimulation();
  return (tokenSimulation);
}
```

A ladder object, after reading its inputs (by sending simulation events to all its left-connections), must call its own *OwnerSimulation* method in order to simulate the instruction it represents. In this work, the *OwnerSimulation* methods simulate the behaviour of the corresponding instructions of a real PLC (OMRON C20H [7]). As an example, the following listing presents the *OwnerSimulation* methods for *CloseContact* and *OpenOutput* classes.

```
BOOL CloseContact :: OwnerSimulation()
{
flagSimulation = TRUE,
if( tokenSimulation )
  return (tokenSimulation = (*memLocation & maskBit) ? FALSE : TRUE );
  return (tokenSimulation = FALSE );
}
```

```
BOOL OpenOutput::OwnerSimulation()
{
flagSimulation = TRUE;
if(tokenSimulation)
  *memLocation |= maskBit;
  else
  *memLocation &= ~maskBit;
return (tokenSimulation);
}
```

With the methodology presented here, it is not difficult to add new instructions to the virtual PLC. To implement a new instruction it is necessary to derive a class from the hierarchy of ladder objects; the virtual method *Simul* should, if necessary, be redefined (except for single cell objects, because of the inheritance mechanism of object oriented techniques); the *OwnerSimulation* method must be redefined for each new ladder object since it is responsible for the particular action of each instruction.

4. CONCLUSIONS

It is well known that simulation is of vital importance in several distinct fields, namely in the industrial automation area. Also accepted are the advantages of object oriented techniques, in the development of simulation algorithms.

In this work, a generic and distributed algorithm was implemented to simulate relay ladder logic language. Generic, because it is independent of the ladder objects (instructions) to be simulated, and distributed, because each instruction, represented by an object, has its own simulation routine. This allows future upgrading without changes in the simulator, even with the introduction of new instructions in the language.

The validation of the simulation algorithm has been done by running several programs, with both the virtual PLC and a real PLC.

5. REFERENCES

1. Hughes, T. A., Programmable Controllers, ISA, 1989.

2. Picard, R. P. and Savage, G. J., "Instructing Industrial Controls Using Ladder Diagrams on an IBM PC", IEEE Transactions on Education, vol. E-29, no. 1, 1986.

3. Ferreira, J. A., Virtualization of Programmable Logic Controllers, Master Thesis. U. Aveiro, Aveiro (Portugal), 1994.

4. Stroustrup, B., The C++ programming language, Addison-Wesley, 1986.

5. Bischak, D. P. and Roberts, S. D., "Object Oriented Simulation", Proceedings of the 1991 Winter Simulation Conference, Phoenix (USA), 1991.

6. Estima, J. O., Azevedo, J. L., Ferreira, J. A. and Ferreira, P. J., "Software Development for Programmable Logic Controllers - a methodology and a system", Proceedings of the IFAC Workshop on CIM in Process and Manufacturing Industries, Espoo (Finland), 1992.

7. OMRON Corporation, Mini H-type Programmable Controllers - Operation Manual, 1990.

Designing Automated Reasoners with Efficient Algorithms

J. W. Guan, D. A. Bell
School of Information and Software Engineering
University of Ulster at Jordanstown
BT37 0QB, Northern Ireland, U.K.
E-mail: {J.Guan,DA.Bell}@ulst.ac.uk
Tel: +44 1232 368913/366674
Fax: +44 1232 366068

ABSTRACT

An intelligent system should take control according to the information it gets. It also should change its control according to the changing information. We assume that in this system, there is a set of nodes. Each node can get a piece of evidence. It is a 3-tuple (p, c, r), where $p+c+r = 1$ and p is the measure of support for the node; c is the measure of support for the refutation of the node; r is the measure of the support not assigned for or against the node. We present efficient algorithms to collecting evidence; i.e., algorithms combining and updating by the Dempster-Shafer theory of evidence.

Keywords: Artificial Intelligence, Intelligent Systems, Algorithms, Evidence Theory

1 Introduction

An intelligent system should take control according to the information it gets. It also should change its control according to the changing information. Here we present a figure of intelligent systems as follows.

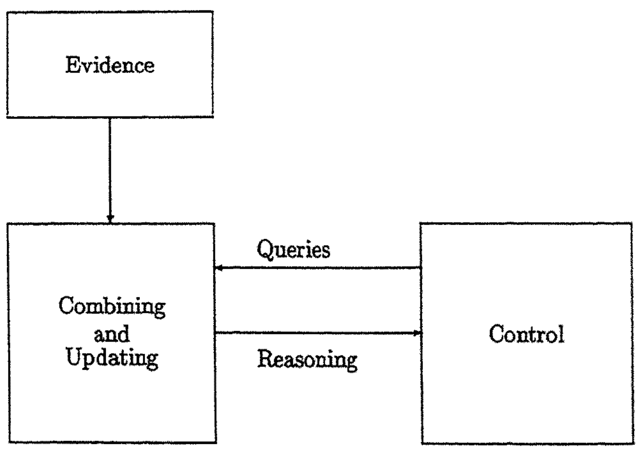


Figure 1. An Intelligent System

More precisely, we assume that in this system, there is a set of nodes $\Theta = \{x_1, x_2, ..., x_i, ..., x_{|\Theta|}\}$. Suppose that for every $i = 1, 2, ..., n, ..., |\Theta|$, there is a piece of evidence m_i (so-called dichotomous mass function). Every piece of evidence m_i is given by 3 values: $m_i = (p_i, c_i, r_i)$, where $p_i + c_i + r_i = 1$ for $i = 1, 2, ..., n, ..., |\Theta|$, and

1. (*Positive:*) p_i is the measure of support for node $\{x_i\}$, or $pro(\{x_i\})$;

2. (*Negative:*) c_i is the measure of support for the refutation of node $\{x_i\}$, or $con(\{x_i\})$; and

3. (*Remain unknown:*) r_i is the measure of the support not assigned for or against the node $\{x_i\}$, or remaining after $pro(\{x_i\})$ and $con(\{x_i\})$ have been established.

We will present efficient algorithms to collecting evidence; i.e., algorithm combining and updating by the Dempster-Shafer theory of evidence. Section 2 introduces evidence theory. Section 3 introduces the evidence combining algorithm. Section 4 introduces the evidence updating algorithm.

2 Mass Functions and the Dempster-Shafer Rule

Let Θ be a finite non-empty set and call it the *frame of discernment*. This is often taken to be a set of contenders or hypotheses from which we want to make a choice. Let $[0,1]$ denote the interval of real numbers from zero to one, inclusive: $[0,1] = \{x | 0 \leq x \leq 1\}$.

A function $m : 2^{\Theta} \rightarrow [0,1]$ is called a *mass function* if it satisfies

m1) $m(\emptyset) = 0$,

m2) $\sum_{X \subseteq \Theta} m(X) = 1$.

A mass function is a *basic probability assignment* to all subsets X of Θ.

The fundamental operation of evidential reasoning is known as the Dempster-Shafer rule for combining evidence. We now define this rule. Let m_1 and m_2 be mass functions on the same frame Θ. Suppose $E = \sum_{X \cap Y = \emptyset} m_1(X) m_2(Y) < 1$. Denote $N = \sum_{X \cap Y \neq \emptyset} m_1(X) m_2(Y)$. Then the function $m : 2^{\Theta} \rightarrow [0,1]$ defined by

1) $m(\emptyset) = 0$, and

2) $m(A) = (1/N) \sum_{X \cap Y = A} m_1(X) m_2(Y)$ for all subsets $A \neq \emptyset$ of Θ

is a mass function. The mass function m is called the *orthogonal sum* of m_1 and m_2, and is denoted $m_1 \oplus m_2$. If $N = \sum_{X \cap Y \neq \emptyset} m_1(X) m_2(Y) = 0$, then we say that the orthogonal sum $m_1 \oplus m_2$ *does not exist* and that m_1 and m_2 are *totally contradictory*. Generally, $K = 1/N$ is called the *normalization constant* of the orthogonal sum of m_1 and m_2. The normalization constant $K = 1/N$ measures the extent of conflict between the two mass functions.

2.1 The Barnett Structure

Because evidence theory is based on sets and enumeration, in general, the computational complexity of evaluating evidential functions is exponential. We explain which we mean by this expression in detail later, but essentially it means that it is not feasible to translate evidence theory into an efficient direct implementation. However, fortunately this computational difficulty is remand completely if the type of evidence combined has a particular structure. The method can be used in any domain where separate pieces of evidence support a singleton proposition or its negation.

Barnett (1981) proposed a method of partitioning the problem space in several independent ways and clustering evidence in the partitions. This allows the evidence to be evaluated in linear time. Here we describe a particularly useful structuring of evidence which was introduced by Barnett in 1981. Generally, a mass function m is said to be *dichotomous* if the only possible focuses of m are $A, \Theta - A, \Theta$ for some $A \subseteq \Theta$. A special case occurs when A is singleton, and in Barnett's structure, we are only interested in dichotomous mass functions m which have no focuses other than $\{x\}, \Theta - \{x\}, \Theta$ for some $x \in \Theta$.

Let $\Theta = \{x_1, x_2, ..., x_i, ..., x_{|\Theta|}\}$. Suppose that for every $i = 1, 2, ..., n, ..., |\Theta|$, there is a dichotomous mass function m_i: $p_i = m_i(\{x_i\}), c_i = m_i(\Theta - \{x_i\}), r_i = m_i(\Theta)$, where $p_i + c_i + r_i =$

1 for $i = 1, 2, ..., n, ..., |\Theta|$. Actually, we know that p_i is the measure of support for $\{x_i\}$, or $pro(\{x_i\})$; c_i is the measure of support for the refutation of $\{x_i\}$, or $con(\{x_i\})$; and r_i is the measure of the support not assigned for or against the proposition $\{x_i\}$, or remaining after $pro(\{x_i\})$ and $con(\{x_i\})$ have been established. Barnett's technique is based on the use of dichotomous mass functions instead of general mass functions. This means that instead of potentially calculation we need only make 3 computations. The dichotomous mass functions concentrate on only the 3 particular subsets (the focal elements) for each $x \in \Theta$ while the general mass functions have to enumerate all $2^{|\Theta|}$ subsets of Θ. So we seek to present efficient methods to compute the quantities associated with the orthogonal sum of n ($n \geq 1$) dichotomous mass functions.

2.2 Formulae for the Repeated Focuses

Let $m_1, m_2, ..., m_n$ be n dichotomous mass functions with the same dichotomy on 2^Θ:

$$p_i = m_i(\{x\}), c_i = m_i(\Theta - \{x\}), r_i = m_i(\Theta), m_i(elsewhere) = 0$$

for $i = 1, 2, ..., n$. Then $m_1, m_2, ..., m_n$ is combinable if and only if $1/K = N > 0$, where

$$1/K = N = p_1 \prod_{i=2}^{n}(p_i + r_i) + c_1 \prod_{i=2}^{n}(c_i + r_i) + r_1 p_2 \prod_{i=3}^{n}(p_i + r_i)$$

$$+ r_1 c_2 \prod_{i=3}^{n}(c_i + r_i) + r_1 r_2 p_3 \prod_{i=4}^{n}(p_i + r_i) + r_1 r_2 c_3 \prod_{i=4}^{n}(c_i + r_i) + ... + \prod_{i=1}^{n} r_i.$$

If $N > 0$, then we have the following (Guan & Bell 1995): The orthogonal sum $m = m_1 \oplus m_2 \oplus ... \oplus m_n$; $1 \leq n \leq |\Theta|$ of n dichotomous mass functions $m_1, m_2, ..., m_n$ is again dichotomous with the same dichotomous. The formula for m on 2^Θ is the following: $m(\emptyset) = 0$.

$$m(\{x\}) = p = K(p_1 \prod_{i=2}^{n}(p_i + r_i) + r_1 p_2 \prod_{i=3}^{n}(p_i + r_i) + r_1 r_2 p_3 \prod_{i=4}^{n}(p_i + r_i)$$

$$+ ... + r_1 r_2 ... r_{n-2} p_{n-1}(p_n + r_n)),$$

$$m(\Theta - \{x\}) = c = K(c_1 \prod_{i=2}^{n}(c_i + r_i) + r_1 c_2 \prod_{i=3}^{n}(c_i + r_i) + r_1 r_2 c_3 \prod_{i=4}^{n}(c_i + r_i)$$

$$+ ... + r_1 r_2 ... r_{n-2} c_{n-1}(c_n + r_n)),$$

$$m(\Theta) = r = K \prod_{i=1}^{n} r_i, m(elsewhere) = 0.$$

2.3 Combining Two Evidence with the Repeated Focuses

Here we are specially interested in the case $n = 2$. Let m_1, m_2 be 2 dichotomous mass functions with the same dichotomy on 2^Θ:

$$p_1 = m_1(\{x\}), c_1 = m_1(\Theta - \{x\}), r_1 = m_1(\Theta), m_1(elsewhere) = 0$$

$$p_2 = m_2(\{x\}), c_2 = m_2(\Theta - \{x\}), r_2 = m_2(\Theta), m_2(elsewhere) = 0.$$

Then m_1, m_2 is combinable if and only if $1/K = N > 0$, where

$$1/K = N = p_1(p_2 + r_2) + c_1(c_2 + r_2) + r_1 p_2 + r_1 c_2 + r_1 r_2.$$

If $N > 0$, then we have the following: The orthogonal sum $m = m_1 \oplus m_2$ of 2 dichotomous mass functions m_1, m_2 is again dichotomous with the same dichotomous. The formula for m on 2^Θ is the following: $m(\emptyset) = 0$.

$$m(\{x\}) = p = K(p_1(p_2+r_2)+r_1p_2) = \frac{p_1(p_2+r_2)+r_1p_2}{p_1(p_2+r_2)+c_1(c_2+r_2)+r_1p_2+r_1c_2+r_1r_2},$$

$$m(\Theta-\{x\}) = c = K(c_1(c_2+r_2)+r_1c_2) = \frac{c_1(c_2+r_2)+r_1c_2}{p_1(p_2+r_2)+c_1(c_2+r_2)+r_1p_2+r_1c_2+r_1r_2},$$

$$m(\Theta) = r = Kr_1r_2 = \frac{r_1r_2}{p_1(p_2+r_2)+c_1(c_2+r_2)+r_1p_2+r_1c_2+r_1r_2},$$

$$m(elsewhere) = 0.$$

3 An Algorithm for Combining All Evidence

ALGORITHM C. (*Combining all ($|\Theta|$ pieces of) dichotomous evidence in the Barnett structure.*)

Let $\Theta = \{x, y, ..., w, ..., z\}$. Suppose that for each $x \in \Theta$ there is a dichotomous mass function as follows: $m_x(\{x\}) = p_x, m_x(\Theta - \{x\}) = c_x, m_x(\Theta) = r_x; 0 \le p_x, c_x, r_x \le 1;\ p_x + c_x + r_x = 1$.

This algorithm checks if there exists the orthogonal sum $m = m_x \oplus m_y \oplus ... \oplus m_z$, and computes it in the existent case.

We use $3|\Theta|$ auxiliary variables P_x, C_x, R_x for every $x \in \Theta$; 2 auxiliary variables Q, S; and $2^{|\Theta|} - 1$ auxiliary variables MA for every subsets A such that $\emptyset \subset A \subseteq \Theta$.

Variables P_x, C_x, R_x are used to deposit evidence p_x, c_x, r_x; respectively.

Variables Q, S are used to deposit $\sum_{x\in\Theta} \frac{p_x}{1-p_x}, \prod_{x\in\Theta} \frac{c_x}{1-p_x}$; respectively.

Upon successful termination of this algorithm, the values of MA give $m(A)$ for every subset $A: \emptyset \subset A \subseteq \Theta$.

C1. [*Initialize.*] Set

$$MA \leftarrow 0 \text{ for all } A \neq \Theta, M\Theta \leftarrow 1;$$

$$P_x \leftarrow p_x, C_x \leftarrow c_x, R_x \leftarrow r_x; P_y \leftarrow p_y, C_y \leftarrow c_y, R_y \leftarrow r_y; ......; P_z \leftarrow p_z, C_z \leftarrow c_z, R_z \leftarrow r_z;$$

$$Q \leftarrow \sum_{x\in\Theta} \frac{P_x}{1-P_x}; S \leftarrow \prod_{x\in\Theta} \frac{C_x}{1-P_x}.$$

C2. [$p_x = 1$ *for* x?] If $P_x = 1$ for exactly one x, then (the orthogonal sum m is successfully computed as follows: $m(\{x\}) = 1, m(elsewhere) = 0$ and to be output) set "x-mass" $M\{x\} \leftarrow 1, M\Theta \leftarrow 0$; output x mass $m = m_x \oplus m_y \oplus ... \oplus m_z$.

Go to U1 to collect new evidence.

Otherwise we need go on the next step to check if $p_x \neq 1$ for all x.

C3. [$p_x \neq 1$ for all x ?] If $P_x \neq 1$ for all x and there exists at least one x such that $C_x \neq 1$, then we need go on the next step to compute m.

Otherwise the orthogonal sum m does not exist.

C4. [*Compute m.*] Set

$$M\{x\} \leftarrow \frac{1}{1+Q-S}\left(\frac{P_x}{1-P_x} + \frac{R_x}{1-P_x}S\right)$$

for all $x \in \Theta$;

$$MA \leftarrow \frac{1}{1+Q-S}\left(S \prod_{x\in A} \frac{R_x}{C_x}\right)$$

for all $A \subseteq \Theta, |A| > 1$.

Output MA for all $\emptyset \subset A \subseteq \Theta$. Go to U1 to collect new evidence. $\diamond$

4 An Algorithm for Updating Evidence in the Barnett Structure

ALGORITHM U. (*Updating evidence when a new piece of) dichotomous evidence comes in the Barnett structure.*) Let $\Theta = \{x, y, ..., w, ..., z\}$. Suppose that for each $x \in \Theta$ there was a dichotomous mass function as follows:

$$m_x(\{x\}) = p_x, m_x(\Theta - \{x\}) = c_x, m_x(\Theta) = r_x; 0 \le p_x, c_x, r_x \le 1;\ p_x + c_x + r_x = 1.$$

Now, suppose that for an element $a \in \Theta$ comes a dichotomous mass function m'_a as follows:

$$m'_a(\{a\}) = p'_a, m'_a(\Theta - \{a\}) = c'_a, m'_a(\Theta) = r'_a; 0 \le p'_a, c'_a, r'_a \le 1;\ p'_a + c'_a + r'_a = 1.$$

This algorithm checks if there exists the orthogonal sum $m = m_x \oplus m_y \oplus ... \oplus (m_a \oplus m'_a) \oplus ... \oplus m_z$, and computes it in the existent case.

We use 3 auxiliary variables P'_a, C'_a, R'_a to deposit the new evidence p'_a, c'_a, r'_a coming from $a \in \Theta$.

We also use $3|\Theta|$ auxiliary variables P_x, C_x, R_x for every $x \in \Theta$; 2 auxiliary variables Q, S; and $2^{|\Theta|} - 1$ auxiliary variables MA for every subsets A such that $\emptyset \subset A \subseteq \Theta$.

Now, variables P'_a, C'_a, R'_a deposit the new evidence p'_a, c'_a, r'_a; respectively. Variables P_x, C_x, R_x deposit the old evidence p_x, c_x, r_x; respectively. Variables Q, S deposit $\sum_{x\in\Theta} \frac{p_x}{1-p_x}, \prod_{x\in\Theta} \frac{c_x}{1-p_x}$; respectively. Variables MA deposit the old values $m(A)$ for every subset $A:\ \emptyset \subset A \subseteq \Theta$.

Upon successful termination of this algorithm, the values of MA give the **updating** $m(A)$ for every subset $A:\ \emptyset \subset A \subseteq \Theta$.

U1. [*Initialize.*] (Now $MA = m(A)$ for all $\emptyset \subset A \subseteq \Theta$ for the old mass function $m_x \oplus m_y \oplus ... \oplus m_a \oplus ... \oplus m_z$;

$$P_x = p_x, C_x = c_x, R_x = r_x; P_y = p_y, C_y = c_y, R_y = r_y; ......;$$

$$P_a = p_a, C_a = c_a, R_a = r_a; ......; P_z = p_z, C_z = c_z, R_z = r_z$$

for the old evidence;

$$Q = \sum_{x\in\Theta} \frac{P_x}{1-P_x}; S = \prod_{x\in\Theta} \frac{C_x}{1-P_x}.$$

for the old evidence.)

Set

$$P'_a \leftarrow p'_a, C'_a \leftarrow c'_a, R'_a \leftarrow r'_a; Q \leftarrow Q - \frac{P_a}{1-P_a}; S \leftarrow S \times \frac{1-P_a}{C_a}.$$

Set

$$M\Theta \leftarrow P_a(P'_a + R'_a) + C_a(C'_a + R'_a) + R_aP'_a + R_aC'_a + R_aR'_a,$$
$$P_a \leftarrow (P_a(P'_a + R'_a) + R_aP'_a)/M\Theta,$$
$$C_a \leftarrow (C_a(C'_a + R'_a) + R_aC'_a)/M\Theta,$$
$$R_a \leftarrow 1 - P_a - C_a.$$

Set

$$Q \leftarrow Q + \frac{P_a}{1-P_a};\quad S \leftarrow S \times \frac{C_a}{1-P_a}.$$

U2. [$p_x = 1$ *for* x?] If $P_x = 1$ for exactly one x, then (the orthogonal sum m is successfully computed as follows: $m(\{x\}) = 1, m(elsewhere) = 0$ and to be output) set "x-mass"

$$M\{x\} \leftarrow 1, M\Theta \leftarrow 0.$$

Output

1. Updating evidence $P_x, C_x, R_x; P_y, C_y, R_y; ...; P_z, C_z, R_z$.
2. Updating x-mass $m = m_x \oplus m_y \oplus ... \oplus (m_a \oplus m'_a) \oplus ... \oplus m_z$.
Go to U1 to collect new evidence.
Otherwise we need go on the next step to check if $p_x \neq 1$ for all x.

U3. [$p_x \neq 1$ for all x ?] If $P_x \neq 1$ for all x and there exists at least one x such that $C_x \neq 1$, then we need go on the next step to compute m.

Otherwise the orthogonal sum m does not exist.

U4. [*Compute m.*] Set

$$M\{x\} \leftarrow \frac{1}{1+Q-S}\left(\frac{P_x}{1-P_x} + \frac{R_x}{1-P_x}S\right)$$

for all $x \in \Theta$;

$$MA \leftarrow \frac{1}{1+Q-S}\left(S \prod_{x \in A} \frac{R_x}{C_x}\right)$$

for all $A \subseteq \Theta, |A| > 1$.

Output 1. Updating evidence

$$P_x, C_x, R_x; P_y, C_y, R_y; ...; P_z, C_z, R_z.$$

2. Updating $m = m_x \oplus m_y \oplus ... \oplus (m_a \oplus m'_a) \oplus ... \oplus m_z$; i.e., output MA for all $\emptyset \subset A \subseteq \Theta$.
Go to U1 to collect new evidence. $\diamondsuit$

5 Summary

An intelligent system should take control according to the information it gets. It also should change its control according to the changing information. We assume that in this system, there is a set of nodes. Each node can get a piece of evidence. It is a 3-tuple (p, c, r), where $p + c + r = 1$ and p is the measure of support for the node; c is the measure of support for the refutation of the node; r is the measure of the support not assigned for or against the node. We present efficient algorithms to collecting evidence; i.e., algorithm combining and updating by the Dempster-Shafer theory of evidence.

References

[1] Barnett, J. A. 1981, " Computational methods for a mathematical theory of evidence ", *Proc. Seventh International Joint Conference on Artificial Intelligence (IJCAI)*, 1981, 868-875.

[2] Buchanan, B. G. ; Shortliffe, E. H. eds. 1984, " Rule-based expert systems: The MYCIN experiments of the Stanford Heuristic Programming Project ", *Reading, MA: Addison-Wesley.*

[3] Guan, J. W. ; Bell, D. A. 1991-2, " Evidence theory and its applications ", Volume 1-2, *Studies in Computer Science and Artificial Intelligence 7-8*, Elsevier, North-Holland.

A DISTRIBUTED ALGORITHM FOR THE SIMULATION OF TEMPERATURE DISTRIBUTION IN METAL CUTTING

C.-H. Lai [†] and S. Law [‡]
[†]School of Computing and Mathematical Sciences, University of Greenwich
Woolwich Campus, Woolwich, London SE18 6PF, U K
[‡]Techworld Asia Ltd
11/F, Unit 11, Kowloon Plaza, 485 Castle Peak Road, Kowloon, Hong Kong

ABSTRACT

Temperature distributions involved in some metal cutting or surface milling processes may be obtained by solving a nonlinear inverse problem. A simplified metal cutting process is considered in this paper. A problem partitioning concept is used to partition the original problem. We propose a distributed algorithm in relation to the choice of numerical schemes in order to simulate the temperature distribution of such problems. Numerical results are provided for three different materials. Possibilities are discussed of encapsulating the algorithm in a computer virtual design environment for metal cutting and extending the algorithm in a real time simulation system.

KEYWORDS

Distributed algorithms, Metal Cutting

1. INTRODUCTION

In intermittent cutting operations such as metal cutting and face milling, the mean tool face temperature has a very important influence on the rate of tool wear and tool life. High temperatures may cause the material to fatigue or deform under face milling or other cutting operations. Therefore accurate simulation of temperature distributions of the piece of metal subject to milling or cutting is vital in order to lengthen the life time of the tool and to guarantee the quality of the cutting. In particular, real-time simulation of such temperature distributions is of industrial interests.

Industrial practices require cutting temperatures experimentally which are then used as empirical data in suitably chosen thermal models. The measurement of physically meaningful temperatures is extremely difficult. However it is possible to assume that the application of a cutting tool at the cutting point is equivalent to the application of a source at the same point. Therefore if one can simulate the equivalent source at the cutting point, one would be able to simulate the unsteady temperature distribution. Such approach is often referred to as an inverse problem approach. The approach is used in this paper to model a simplified cutting process. One aim of this paper is to study a distributed algorithm for the simulation of temperature distributions along metal under cutting operations. Another aim is to examine the possibility of encapsulating the algorithm in a computer virtual design environment.

2. A SIMPLE CUTTING MODEL

Remote sensor methods for the retrieval of temperature distributions were introduced by Lipman et al [3]. Recently, such methods have been developed into more mature inverse methods [1][5][6] for various cutting situations. The use of inverse methods becomes difficult because in most cases, analytical temperature distributions are difficult to derive. In this paper, a sensor located in an neighbourhood of the cutter is included in the mathematical model, and numerical methods are employed instead of analytical methods. Effects of the size of the neighbourhood on the accuracy of the simulated temperature distribution is not discussed in this paper. Without loss of generality, it can be assumed that the neighbourhood is small enough to produce accurate temperature distributions.

To simplify the cutting problem, a piece of metal of infinite length and unit width and of homogeneous material property along the longitudinal direction, is considered. If a cutting tool is applied at a certain position along the width direction, then it is possible to assume a one-dimensional analogy of the physical problem. Assuming the domain of interest along the width is non-dimensionalised, i.e. $0 < x < 1$, and the cutter is applied at $x = x_c$. Then the above cutting problem can be described by the one-dimensional nonlinear parabolic heat conduction model as follows.

$$\frac{\partial \theta}{\partial t} = \frac{\partial}{\partial x}(k(\theta)\frac{\partial \theta}{\partial x}) + Q_c(t)\delta(x - x_c) \tag{1}$$

subject to initial condition $\theta(x,0) = \Theta(x)$ and boundary conditons $\theta(0,t) = \Theta_0$ and $\theta(1,t) = \Theta_1$ where $\theta(x,t)$ is the temperature distribution, $k(\theta)$ is the conductivity of the metal. $Q_c(t)$ is the equivalent source being applied at $x = x_c$, and $\delta(x - x_c)$ is the Dirac delta function.

The equivalent source strength can be obtained by integrating (1) from $x = x_c^-$ to $x = x_c^+$ which gives

$$k(\theta)\frac{\partial \theta}{\partial x}|_{x_c^+} - k(\theta)\frac{\partial \theta}{\partial x}|_{x_c^-} + Q_c(t) = 0 \tag{2}$$

In order to retrieve the source strength, the temperature gradients just to the left ($x = x^-$) and to the right ($x = x^+$) of the cutter must be known. A temperature sensor is attached at $x = x_s$, such that $x_s < x_c < 1$, and let the temperature measured by means of the temperature sensor be $\theta(x_s, t) = T(t)$. The knowledge of the measured temperature at the sensor is then used to back-calculate average or effective measures at the cutting point. Such inverse methods avoid the basic difficulties of the direct method since remote temperatures can be measured more easily and accurately.

For computer simulation purpose, the sensor temperature is modelled by the sinusoidal function $T(t) = \alpha \sin \omega t$. Its maximum value is governed by the amplitude α and its variation within a given period of time is governed by the angular frequency ω. Therefore α is related to the maximum temperature that a piece of metal can sustain in order to maintain a longer life of the cutter or to maintain fine cuts. It is also related to the amount of coolant applied at the cutter in order to maintain a sufficiently low temperature for producing fine cuts. The angular frequency may be used to simulate the rate of coolant applied at the cutter or the speed of the cutter. The relationship between temperature distributions, the parameters α and ω, and the material conductivity is not within the scope of this paper.

3. A PROBLEM PARTITIONING METHOD

In order to solve the inverse problem given in (1) with the extra-condition given at $x = x_s$, the problem is partitioned into three subproblems defined in the subdomains namely, $S_1 = \{x : 0 < x < x_s\}$, $S_2 = \{x : x_s < x < x_c\}$, and $S_3 = \{x : x_c < x < 1\}$. The effect of such partitioning [4] is to remove the unknown source term $Q_c(t)$ in all of the three subproblems associated with the three subdomains, i.e. the differential equations in these three subdomains do not involve the Dirac delta function. Since the temperature is given at $x = 0$ and there is a temperature sensor located at $x = x_s$, therefore Dirichlet boundary conditions are defined at the boundary of S_1. One can then solve the differential equation to obtain the derivative $\frac{\partial \theta}{\partial x}(x_s, t)$. Hence with the knowledge of the temperature $\theta(x_s, t)$ acquired by the temperature sensor at $x = x_s$, an initial value problem can be formulated in S_2. Therefore $\theta(x_c, t)$ can be obtained by solving the initial value problem. Finally another Dirichlet problem can be formulated in S_3. Thus, we have the three subproblems as follow:

$$SP_1: \frac{\partial u_1}{\partial t} = \frac{\partial}{\partial x}(k(u_1)\frac{\partial u_1}{\partial x}) \text{ in } S_1$$

$$\text{subject to } u_1(x,0) = \Theta(x), \quad u_1(0,t) = \Theta_0, \quad u_1(x_s,t) = T(t).$$

$$SP_2: \frac{\partial u_2}{\partial t} = \frac{\partial}{\partial x}\left(k(u_2)\frac{\partial u_2}{\partial x}\right) \text{ in } S_2$$
$$\text{subject to } u_2(x,0) = \Theta(x), \quad u_2(x_s,t) = T(t), \quad \frac{\partial u_2(x_s,t)}{\partial x} = \frac{\partial u_1(x_s,t)}{\partial x}$$

$$SP_3: \frac{\partial u_3}{\partial t} = \frac{\partial}{\partial x}\left(k(u_3)\frac{\partial u_c}{\partial x}\right) \text{ in } S_3$$
$$\text{subject to } u_3(x,0) = \Theta(x), \quad u_3(x_c,t) = u_2(x_c,t), \quad u_3(1,t) = \Theta_1.$$

The above three subproblems are well-defined, and unique solution exists for each of them. The superposition of these subproblem solutions gives the temperature distribution. Analytic temperature distribution is difficult to obtain and only a few simplified cases can be dealt with [5] Instead, a numerical method suitable for distributed computing is proposed.

4. THE DISTRIBUTED NUMERICAL ALGORITHM

One obvious way of distributing subproblems is to employ as many loosely coupled workstations as the number of subproblems. In the present case there should be three workstations It is possible to treat the algorithm as a pipe line process, in which case SP_1 will be solved first and then SP_2, etc. Therefore there is a time-lag for SP_3 compare with SP_2 and SP_1, i.e. all of the three workstations are kept busy at the beginning of the third time step and before the last two time steps. Let W_{SP_i} denotes the computational work involved in solving SP_i. The situation $W_{SP1} \neq W_{SP2} \neq W_{SP3}$ yields a distributed algorithm with computing time depending on $\max\{W_{SP1}, W_{SP2}, W_{SP3}\}$. Since the sensor is located in the neighbourhood of the cutter, therefore the subdomain S_2 is usually very small. Since $u_1(x_s,t) = T(t)$ is given, the subproblem SP_1 can be operated completely independent of SP_2 and SP_3. It is possible to reduce the communication time by putting SP_2 and SP_3 into a workstation for sequential process. In such situation, the computation of the source strength does not involve inter-processor communication.

To maintain load balancing in the two workstations, we require $W_{SP1} \simeq W_{SP2} + W_{SP3}$. The equality means that the two processes are synchronised. If $W_{SP1} \neq W_{SP2} + W_{SP3}$, then it is important that $\frac{\partial u_1(x_s,t)}{\partial x}$ is sent from the first processor to the second processor and is kept in the local memory of the second processor for the use in subsequent computations. For synchronised processes, such storage is not necessary. The distributed algorithm developed in this paper is based on the above approach and is given as follows:

Distributed Algorithm for Metal Cutting Process

Processor 1:
- for i = 1. number_of_steps
  - $t := i * \Delta t$;
  - Compute the solution of SP_1 at t; Compute $\frac{\partial u_1(x_s,t)}{\partial x}$;
  - Non-blocking Send $\frac{\partial u_1(x_s,t)}{\partial x}$ to Processor 2;
- end-for

Processor 2:
- for i = 1, number_of_steps
  - $t := i * \Delta t$;

Blocking Receive $\frac{\partial u_1(x_s,t)}{\partial x}$ from Processor 1;
Compute the solution of SP_2 at t; Compute the solution of SP_3 at t;
Compute $\frac{\partial u_2(x_s,t)}{\partial x}$ and $\frac{\partial u_3(x_s,t)}{\partial x}$; Retreive $Q_c(t)$ using (2);
end-for

The meaning of non-blocking send in the above algorithm is that computation in the sending processor ressumes as soon as the message is safely on its way to the receiving processor. The meaning of blocking receive in the above algorithm is that the receiving processor has to wait until the correct message from the sending processor has arrived.

4.1 Numerical schemes

A first order difference approximation of the temporal derivative and a second order difference approximation of the spatial derivatives are used in the subproblems SP_1 and SP_3. An explicit scheme is resulted from the difference approximation. Dropping the subscripts used in denoting the subdomains. the explicit scheme for SP_1 and SP_3 can be written as

$$u_i^{(n+1)} = rb_i^{(n)}u_{i-1}^{(n)} + (1 - r(a_i^{(n)} + b_i^{(n)}))u_i^{(n)} + ra_i^{(n)}u_{i+1}^{(n)} \tag{3}$$

where i denotes the i-th grid point, $r = \frac{\Delta t}{(\Delta x)^2}$, $a_i^{(n)} = \frac{k_{i+1}^{(n)}+k_i^{(n)}}{2}$, $b_i^{(n)} = \frac{k_i^{(n)}+k_{i-1}^{(n)}}{2}$, (n) denotes the time-step, Δt is the step size along the temporal axis and Δx is the mesh size along the spatial axis x.

The subproblem SP_2 becomes a second order intial value problem along the spatial dimension when the first order backward difference approximation, denoted as m, of the temporal derivative at $x = x_s$ is substitued into $\frac{\partial u_2}{\partial t}$ of SP_2. A one-step modified Euler integration scheme is applied to solve the initial value problem and is written as the following pair of iterations

$$\begin{pmatrix} u^* \\ v^* \end{pmatrix} := \begin{pmatrix} u \\ v \end{pmatrix} + \Delta x \underline{f} \tag{4}$$

$$\begin{pmatrix} u \\ v \end{pmatrix} := \begin{pmatrix} u \\ v \end{pmatrix} + \frac{\Delta x}{2}\left\{\underline{f} + \underline{f}^*\right\} \tag{5}$$

where $v = \frac{\partial u}{\partial x}$, and $\underline{f}^* = \begin{pmatrix} v^* \\ \frac{m}{k} - \frac{k'}{k}v^{*2} \end{pmatrix}$.

4.2 Message Passing Interface

A sequential Fortran program has been written to perform the different tasks as listed in the above algorithm. MPI is used to provide distributive directives in order that the tasks can be distributed onto a network of Sun workstations. For the present studies, only two Sun workstations are required.

MPI or its name in full, Message Passing Interface, is a standard consists of a set of Fortran subroutine calls that allows a heterogeneous collection of workstations and/or supercomputers to function as a single high-performance parallel machine. The environment consists of a loosely coupled workstations and is hidden to users of scientific and engineering packages. The MPI standard began in the summer of 1992 and its version 1.0 standard was ready by 1993. The smallest unit of parallelism in MPI is a task and users write the parallel application as a collection of cooperative tasks. A task is usually designed for sequential machine and can be written in Fortran language. These tasks can be communicated by explicitly sending and receiving messages through the virtual machine using suitable MPI calls [2] to form cooperating tasks. The present application follows the above idea in such a way that communications are done by nonblocking send and blocking receive.

4.3 Numerical tests

A number of tests was performed by taking the conductivity as $k(u) = a + bu + cu^2 + du^3$, where a, b, c, and d are parameters used to describe the material property of a piece of metal. Table 1 shows three sets of different conductivity parameters used in the subsequent tests. The nonlinearity of the conductivity decreases as $|c|$ and $|d|$ decreases. In particular the last set of conductivity parameters gives a constant conductivity. For the conductivity as shown in Table 1, α and ω are chosen to be 0.4 and 2π respectively. The initial value $\Theta(x)$ and the boundary values Θ_0 and Θ_1 are chosen to be zeros. The locations of the sensor and the cutter, i.e. x_s and x_c, are chosen to be 0.4 and 0.5 respectively. Numerical results are provided for two mesh sizes $\triangle x = \frac{1}{20}$ and $\triangle x = \frac{1}{40}$ and the corresponding $\triangle t$'s are chosen to be 0.001 and 0.0002. The resulting numbers of modified Euler steps in SP_2 are two and four respectively.

Material	a	b	c	d
A	0.5	-1.1	1.0	-1.0
B	0.8	-0.5	0.01	-0.01
C	1.0	-0.3	0.0001	-0.0001

Table 1: Conductivity parameters.

Temperature distributions at $t = 1.1$ seconds are shown in Fig 1 and Fig 2. The results show that $\triangle x$ has little effect on the temperature distribution. Similar comparison holds at all other values of t. For materials B and C, the corresponding source strength variation with respect to t is similar for different mesh sizes. For material A, the source strengths for different mesh sizes have a large discreprency at a particular time which comes in periodically. Otherwise the effect of mesh size is again negligible. It is largely due to the nonlinearity introduced into the differential equation. Computationally, the parameter $\alpha = 0.4$ is too low for material A, i.e. a slightly higher value of α can be used, which represents a smaller amount of coolant being applied at the tool face.

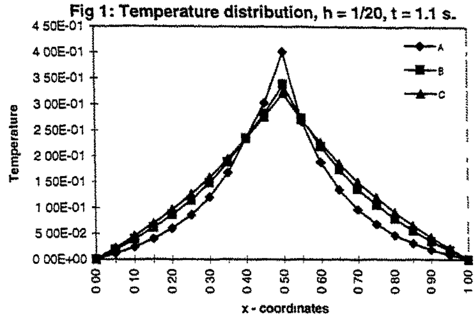
5. VIRTUAL DESIGN ENVIRONMENT

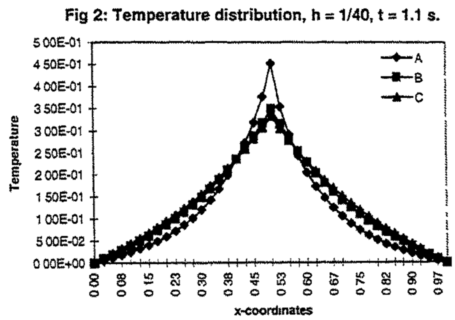
In some industrial applications, it is important to provide fine cut of materials. Therefore real time control of cutter temperatures, cutter speed, and cutter fluid emission is important. In order to achieve real time simulation, it is necessary to reduce the overall computing time which can be done by means of incorporating vector or other vector/parallel processing capabilities to a workstation in order to enhance the computational power.

Another curent development is to encapsulate the algorithm in a virtual design environment. A good graphical capability and user friendly interface is needed for such a design environment. Usually a control panel is needed to input information concerning heat conductivity of the metal, cutter location, the amplitude α, and the angular frequency ω as described in Section 4.2. Another panel which is used to display the temperature distribution of the metal is also vital. Such environment would allow a designer to understand more about the relationship between various parameters and the respective temperature distribution. A preliminary version of the environment is at α-testing and we are working on a 2-D version with enhanced graphics.

6. CONCLUSIONS

The use of the distributed algorithm for the retrieval of heat source at the cutter and the back calculation of the temperature distribution is demonstrated in this paper. MPI is used to provide necessary distributive directives. Fast and parallel numeical schemes for individual subproblems may be incorporated in order to reduce the overall computing time. The algorithm can be encapsulated in a virtual design environment for rapid understanding of temperature





distribution in a cutting process.

7. REFERENCES

1. Chow, J.G. and Wright, P K. On-line estimation of tool/chip interface termperature for a turning operation. *ASME Journal of Engineering for Industry*, **110**, 56 - 64, 1988.

2. Gropp, W., Lusk, E. and Skjellum, A. Using MPI: Portable Parallel Programming with the Message-Passing Interface, MIT Press, 1994.

3. Lipman, M.P., Nevis, B.E. and Kane, G.E. A remote sensor method for determining average thermal properties heated by moving heat sources. *ASME Journal of Engineerng for Industry*, **89**, 333 - 338, 1967.

4. Lai, C.-H. Diakoptics, domain decomposition and parallel computing. *The Computer Journal*, **37**, 840 - 846, 1994.

5. Stephenson, D.A. An inverse method for investigating deformation zone temperatures in metal cutting. *ASME Journal of Engineering for Industry*, **113**, 129 - 136, 1991.

6. Yen, D.W. and Wright, P.K. A remote temperature sensing technique for estimating the cutting interface temperature distribution. *ASME Journal of Engineering for Industry*, **108**, 252 - 263, 1986.

FUZZY SELF ORGANIZING CONTROL FOR CONTOURING PERFORMANCE IMPROVEMENT OF CNC MACHINE TOOLS

J. John Cheng
Science Center, Rockwell International
Thousand Oaks, CA 91360, U.S.A.

Tom Kong and Daniel C. H. Yang
Mechanical and Aerospace Engineering Department
University of California, Los Angeles
LA, CA 90204, U.S.A.

ABSTRACT

The objective of this study was aimed at reducing the contour errors of CNC machines by improving the tracking performance of each axis through on-line adaptation and nonlinear fuzzy control of an effective adaptive fuzzy control method, Fuzzy Self Organizing Control (FSOC). All reported FSOC applications employed a fuzzy controller with two inputs, error and change-in-error. Observed in a laboratory experiment, simple-minded applying FSOC in this conventional 2-dimension fuzzy control implementation resulted in excessive adaptation transient during the phase changes of cutting path command, causing the tripping of the motor drive. The cause of the excessive adaptation transient was analytically investigated, leading to an improved FSOC implementation. The improved FSOC employs a high-dimension fuzzy controller which incorporates the cutting path command information. Demonstrated on a 2-axis motion control test bed, the high-dimension FSOC achieved significantly improved contour accuracy compared to a traditional velocity feedforward plus PI (Proportional + Integral) controller.

KEYWORDS

Adaptive Fuzzy Control, Fuzzy Self Organizing Control, CNC, Contour Error

1. INTRODUCTION

Due to mismatch in axis dynamics, load disturbance, and nonlinearities, contour errors occur in conventional CNC machines [7]. One approach to reduce the contour errors is to improve the dynamic performance of each individual axis. The rationale behind this approach is that zero tracking error of each axis translates to zero contour error. One method for improving individual axis tracking performance is the application of feedforward control. In industrial CNCs, simple velocity or velocity plus acceleration feedforward control is widely used. Advanced feedforward control algorithms such as Zero Phase Error Tracking (ZPET) [14] and Modified Inverse Transfer Function (MITF) [9] are mainly used in research prototypes. A major drawback of feedforward control lies in its inability to adapt to modeling errors and parameter deviations. Pritschow and Philipp [9] reported that under the condition of +-5% parameter uncertainty, the use of velocity plus acceleration feedforward control, ZPET, or MITF did not improve the contour accuracy. In this work, we explored adaptive control approach, specifically, adaptive fuzzy control, to improve tracking performance and in turn achieve better contour accuracy under model and parameter uncertainties.

One effective adaptive fuzzy control design is the Fuzzy Self Organizing Control (FSOC) proposed in [10]. Successful applications were reported, for example, in [1,2,3,4,5,6,8,11,12]. The objective of this study was to extend and capitalize the learning and nonlinear fuzzy control strengths of FSOC toward CNC application, emphasizing on contour performance improvement under modeling errors and parameter deviations. This application presented a non-trivial challenge. It was observed in a laboratory experiment that for CNC control, the simple-minded applying FSOC in its conventional 2-dimension fuzzy control implementation (to be discussed in Section 3) resulted in excessive adaptation transient during the phase changes between acceleration, constant feedrate, or deceleration of the cutting path command, eventually causing the tripping of the motor drive due to over-current fault.

In the following, we first briefly review the concept of FSOC in Section 2. In Section 3 we investigate the reason why the conventional FSOC caused excessive adaptation transient during the phase changes of the cutting path command. The investigation motivates an extension of the conventional FSOC design. The concept of the extended FSOC design is presented in Section 4. The effectiveness of the extended FSOC solution has been validated on a 2-axis motion control test bed and compared to a velocity feedforward plus PI controller. The experiment results are reported in the same section. A conclusion is stated in Section 5.

2. A BRIEF REVIEW OF FUZZY SELF ORGANIZING CONTROL

The concept of FSOC is briefly reviewed here assuming a knowledge of fuzzy logic. For details, the reader is referred to [3,10,11,12] for FSOC and [13] for fuzzy logic. The concept of FSOC design was motivated by human experience-organizing and decision-making processes (Figure 1, [10]). It includes two fuzzy systems: Fuzzy Controller and Performance Index. Analytically speaking, it is a Model Reference Adaptive Control (MRAC, Figure 2) with nonlinear adaptation mechanism implemented through Performance Index and a nonlinear feedback controller represented in fuzzy logic [3]. Different implementations of FSOC are possible, depending upon what fuzzy controller's parameters are adjusted. In this section, we review one implementation in which the controller output is linearly dependent on the unknown parameters of the controller [3]. The advantage of this implementation is its tractability. In the following, we first define the unknown parameters (θ) of the Fuzzy Controller and then derive its update law ($\Delta\theta[n]$) by describing the input-output functions of the individual blocks shown in Figure 1.

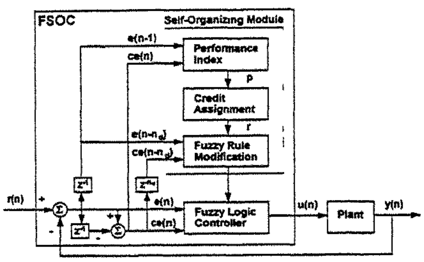


Figure 1. Basic Structure of FSOC

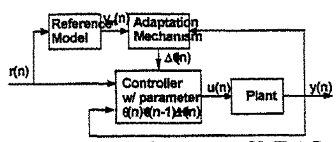


Figure 2. Basic Structure of MRAC

2.1. Fuzzy Logic Controller with Unknown Parameters

In this study, a simplified Takagi-Sugeno model for fuzzy system is used [13]. A typical fuzzy rule for two input variables, error (e) and change-in-error (ce), is:

$$\text{IF } e \text{ is } E_i \text{ and } ce \text{ is } CE_i, \text{ THEN } u = a_i \qquad (1)$$

where E_i and CE_i are fuzzy sets with membership functions $\mu_{Ei}(.)$ and $\mu_{CEi}(.)$, u is the output variable, and a_i is a real number. It is assumed that a_i's are unknown but $\mu_{Ei}(.)$ and $\mu_{CEi}(.)$ are known.

Given a real-valued input vector (e[n], ce[n]) at time n, the output u[n] is calculated as:

$$u[n] = \left(\sum_{i=1}^{N_r} w_i a_i\right) \Big/ \left(\sum_{i=1}^{N_r} w_i\right) = \Psi^T(e[n], ce[n])\,\theta \quad (2)$$

where Nr is the total number of fuzzy rules, $w_i[n] = \min(\mu_{Ei}(e[n]), \mu_{CEi}(ce[n]))$, and

$$\Psi^T = \left(w_1 \Big/ \sum_{j=1}^{N_r} w_j, \ldots, w_{N_r} \Big/ \sum_{j=1}^{N_r} w_j\right) \quad (3)$$

$$\theta^T = (a_1, a_2, \cdots\cdots, a_{N_r}) \quad (4)$$

Note that the output, u, is linearly dependent on the unknown parameter vector θ.

2.2. Parameter Adaptation

The parameter adaptation process involves three elements: Performance Index, Credit Assignment, and Rule Modification (see Figure 1 above).

2.2.1. Performance Index

The Performance Index is also a fuzzy system with input (e[n], ce[n-1]) and output p[n] at time n. The output p[n] is calculated in the same way as described above. The rationale of Performance Index is best illustrated via example. Assume the fuzzy logic map of a typical Performance Index fuzzy system is

ce[n-1] \ e[n]	NegBig	NegSml	Zero	PosSml	PosBig
PosBig	0	+1	+2	+3	+4
PosSml	-1	0	+1	+2	+3
Zero	-2	-1	0	+1	+2
NegSml	-3	-2	-1	0	+1
NegBig	-4	-3	-2	-1	0

Table 1: A typical performance index of a FSOC controller

where {NegBig, NegSml, Zero, PosSml, PosBig} are fuzzy set values of e and ce with linguistic meanings as those commonly interpreted. The zero entries in the map indicate the ideal performance trajectory of an embedded reference model. The nonzero entries provide a measure on the degree in which the performance of the controlled plant deviates from the ideal one.

2.2.2. Credit Assignment

Credit Assignment module uses the simple gradient optimization approach to assign the correction amount, r[n], to the controller output:

$$r[n] = \rho \left(\frac{\partial y[n]}{\partial u[n - n_d]}\right) p[n] \quad (5)$$

where ρ is an adaptation coefficient, p[n] the output of Performance Index, and n_d the time delay of the plant.

2.2.3. Rule Modification

Since the control action is the combined result of individual fuzzy rules, the amount of change of each rule is modulated by its truth value, w_i:

$$a_i[n] = a_i[n\text{-}1] + \left(\frac{w_i[n - n_d]}{\sum_{j=1}^{N_r} w_j[n - n_d]}\right) r[n] \quad (6)$$

Applying Eq (3), Eq (6) can be expressed in vector form:

$$\theta[n] = \theta[n\text{-}1] + \Psi(e[n - n_d], ce[n - n_d])\, r[n] \quad (6')$$

2.2.4. Parameter Adaptation Law of FSOC

Combining the equations (2), (5), and (6) or (6'), we have the parameter adaptation law of FSOC:

$$a_i[n] = a_i[n-1] + \rho \left(\frac{w_i[n-n_d]}{\sum_{j=1}^{N_r} w_j[n-n_d]} \right) \left(\frac{\partial y[n]}{\partial u[n-n_d]} \right) p[n] \qquad (7)$$

or in vector form

$$\theta[n] = \theta[n-1] + \rho\, \Psi(e[n-n_d], ce[n-n_d]) \left(\frac{\partial y[n]}{\partial u[n-n_d]} \right) p[n] \qquad (7')$$

3. DECIPHERING THE CAUSE OF EXCESSIVE ADAPTATION TRANSIENT OF CONVENTIONAL FSOC FOR CNC MACHINE

3.1. Tracking Error in CNC Control

For motion and CNC control, it is well known that under modeling and parameter uncertainties, a velocity plus acceleration feedforward together with a P (Proportional) or PD (Proportional + Integral) feedback control will have non-zero steady-state or divergent tracking errors for constant feedrate or constant acceleration cutting path command respectively Assume that the dynamics of the motor drive is described as a first order system followed by an integrator:

$$H(s) = \frac{k_0}{s(\tau_0 s + 1)} \qquad (8)$$

Without loss of generality, consider a closed-loop P controlled drive with velocity and acceleration feedforward. Denote the transfer function of P control as $G_{FB}(s) = k_p$ and that of feedforward as $G_{FF}(s) = k_a s^2 + k_v s$. The transfer function of closed-loop controlled drive with feedforward becomes:

$$H_{close}(s) = \frac{H(s)G_{FB}(s) + H(s)G_{FF}(s)}{1 + H(s)G_{FB}(s)}$$

$$= \frac{k_0 k_a s^2 + k_0 k_v s + k_0 k_p}{\tau_0 s^2 + s + k_0 k_p} = 1 + \frac{(k_0 k_a - \tau_0)s^2 + (k_0 k_v - 1)s}{\tau_0 s^2 + s + k_0 k_p} \qquad (9)$$

Equation 9 clearly shows the existence of no-zero tracking error when there are uncertainties in motor drive dynamics. In order to achieve zero tracking error, an extra feedforward compensator with transfer function, $G_{extra}(s)$, is needed:

$$G_{extra}(s) = \left(\frac{\tau_0}{k_0} - k_a\right)s^2 + \left(\frac{1}{k_0} - 1\right)s \qquad (10)$$

The amount of extra feedforward command required for zero tracking error performance following a typical cutting profile (Figure 3a) can be calculated and is shown in Figure 3b. The important point in Figure 3b is the discontinuity in the required extra feedforward command.

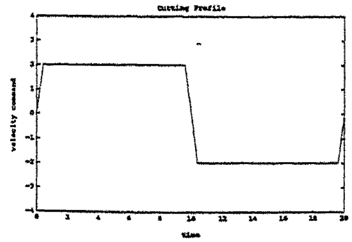


Figure 3(a). Conventional Cutting Profile

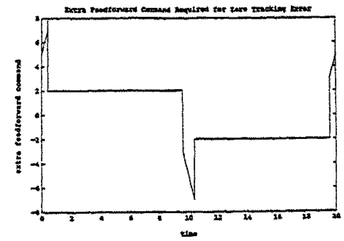


Figure 3(b). Required Extra Feedforward Command (For illustration purpose, $(\tau_0/k_0 - k_a) = (1/k_0 - k_v) = 1$)

3.2. Conventional FSOC for CNC Control

The above review of tracking errors suggests that when applied to CNC control, in order to achieve zero-tracking error the functionalities of an FSOC include not only providing suitable feedback control but also providing an extra feedforward compensation when there are model uncertainties. All reported FSOC applications employed a fuzzy controller with only two inputs, error and change-in-error, which can be considered as a nonlinear PD controller. For feedforward compensation, appropriate compensation amount is reflected in a nonzero value at the center (i.e., zero-zero) cell of the fuzzy map of the fuzzy controller.

It was observed at laboratory that when there were drive model uncertainties, the nonzero value at the zero-zero cell changed constantly in a trend following the prediction by Eq. (10). Furthermore, because of the discontinuity in the required extra feedforward compensation at the phase change transition (Figure 3b) and the unavoidable finite adaptation time of FSOC, the phase-plane trajectory of error and change-in-error jumped away from the center of the phase plane. Instead of attributing such sudden phase-plane trajectory jump to the discontinuity in the required extra feedforward compensation at zero-zero cell, conventional FSOC incorrectly adjusted control parameters of fuzzy rules which were far away from the zero-zero cell of the fuzzy map. The incorrect attribution by FSOC caused vibrant oscillation of the motor, eventually resulting in the tripping of the drive due to over-current fault.

4. FSOC WITH HIGH-DIMENSION (>2) FUZZY FEEDBACK CONTROL - EXPERIMENT RESULTS

To overcome the above over-current fault issue, an FSOC with a high-dimension fuzzy controller is proposed. The high-dimension fuzzy controller incorporates the cutting path command information along with error and change-in-error signals.

4.1. Evaluation and Comparison of Contour Performance

The effectiveness of the high-dimension FSOC was evaluated on a motion control test bed. The test bed consists of two independently controlled DC motors (Figure 4), emulating a two-axis contour machine but without the effect of the cross-coupling mechanism.

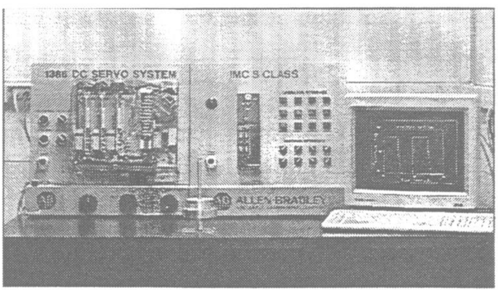


Figure 4. Experiment Setup

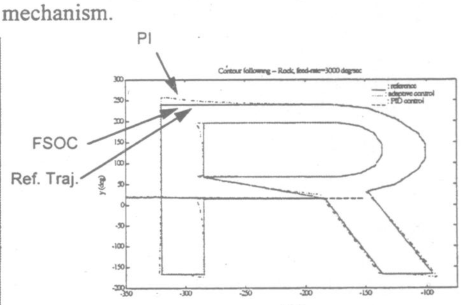


Figure 5. Contour Performance Comparison between FSOC and PI control

The implemented CNC control system included a linear interpolator, a velocity feedforward control, and a feedback controller. Two feedback controllers were implemented: the proposed high-dimension FSOC and a manually tuned PI controller. A 20% parameter error was injected into the experiment by setting the velocity feedforward gain to 80% of the ideal value. Figure 5 showed that the high-dimension FSOC successfully learned to compensate the parameter error and achieved a near perfect contour accuracy. In contrast, poor contour performance of the PI controller was observed, especially around the corners.

5. CONCLUSION

In this paper, we studied the applicability of Fuzzy Self Organizing Control (FSOC) to CNC machines, emphasizing on reducing the contour error through on-line adaptation. Observed in a laboratory experiment, for CNC control simple-minded application of FSOC in its conventional implementation resulted in excessive adaptation transient, causing the tripping of the motor drive. The cause of the excessive adaptation transient was analytically investigated by examining the tracking error in individual axis when there are uncertainties in motor drive dynamics. Motivated by the investigation result, a new FSOC implementation with a high-dimension fuzzy feedback controller was proposed. The effectiveness of the proposed high-dimension FSOC was evaluated on a 2-axis motion control test bed. The new FSOC achieved significantly improved contour accuracy compared to a traditional velocity feedforward plus PI controller.

6. REFERENCES

[1] Cheng, J. J., Ru, R. W., Pham, H. K. and Larson, C. R., "Fuzzy self organizing control for flexible structures with non-collocated sensor and actuator," *Proc. 2nd Int'l Conf. Fuzzy Theory and Technology, Durham, N. Carolina*, Sep. 1993, pp. 182-184.

[2] Cheng, J. J., Ru, R. W., Pham, H. K. and Larson, C. R., "Kinematics and control of redundant manipulators," *Technical Report, Rockwell International Inc. - Science Center*, May 16, 1994.

[3] Cheng, J. J. and Kong, C. T., "Fuzzy self organizing control - theory and application," *7th Annual Rockwell Conf. on Control and Signal Processing*, Thousand Oaks, CA, May 24-25, 1995.

[4] Daley, S. and Gill, K. F., "Comparison of a fuzzy logic controller with a P+D control law," *Journal of Dynamic Systems, Measurement, and Control*, Vol. 111, June 1989, pp. 128-137.

[5] Farbrother, H. N., Stacey, B. A. and Sutton, R., "Fuzzy self-organizing control of a remotely operated submersible," *International Conf. on Control*, Edinburg, Scotland, Mar. 25-28, 1991, pp. 499-504.

[6] Jee, S. and Koren, Y., "A self-organizing fuzzy logic control for friction compensation in feed drives," *Proc. of American Control Conference*, Seattle, WA, June 1995, pp. 205-209.

[7] Koren, Y. and Lo, Ch.-Ch., "Variable-gain cross-coupling controller for contouring," *Annals of the CIRP*, Vol. 40, 1991, pp. 371-374.

[8] Nguyen, H. T. and Sands, D., "Real-time self-organizing fuzzy logic controller for DC servo," in *Fifth European Conf. on Power Electronics and Application*, Vol. 4, 13-16 September 1993, pp. 174-179

[9] Pritschow, G. and Philipp, W., "Research on the efficiency of feedforward controllers in M direct drives," *Annals of the CIRP*, Vol. 41, 1992, pp. 411-416.

[10] Procyk, T. J. and Mamdani, E. H., "A linguistic self-organizing process controller," *Automatica*, Vol. 15, 1979.

[11] Scharf, E. M. and Mandic, N. J., "The application of a fuzzy controller to the control of a multi-degree-of-freedom robot arm." in *Industrial Applications of Fuzzy Control*, M. Sugeno, ed., Elsevier Science Publishing Company, New York, 1985.

[12] Shao, S., "Fuzzy self-organizing controller and its application for dynamic processes," *Fuzzy Sets and Systems*, Vol. 26, 1988, pp. 151-164.

[13] Takagi, T. and Sugeno, M., "Fuzzy identification of systems and its applications to modelling and control," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 15, 1985, pp. 116-132.

[14] Tomizuka, M., "Zero phase error tracking algorithm for digital control," *ASME J. of Dynamic Systems, Measurement, and Control*, Vol. 109, March 1987, pp. 65-68.

SWEPT VOLUME APPROACH AS AN INTEGRAL PART OF 5-AXIS NC MACHINING CAD/CAM SYSTEM

L.P. Wang, M.C. Leu, and *D. Blackmore
Department of Mechanical Engineering
***Department of Mathematics**
New Jersey Institute of Technology
Newark, New Jersey 07102, USA

ABSTRACT

This paper presents a research project developed at the Robotics and Intelligent Manufacturing Laboratory of New Jersey Institute of Technology. The integral swept volume study with part design, multi-axis NC tool path planning, NC program verification and actual NC machining is described. An example on design, verification and manufacturing of a turbine impeller using the integrated CAD/CAM system of 5-axis NC milling is given to illustrate the approach and the result.

KEYWORDS

CAD/CAM, Multi-axis NC Machining, Swept Volume

1. INTRODUCTION

With the rapid development of CAD/CAM technologies, many state-of-the art computer aided design and manufacturing software packages have been developed to design a part, generate tool path, conduct postprocessing, simulate and detect common errors in NC programs, etc. Swept volume is one of the most important concepts associated with CAD/CAM technologies. It has been broadly applied in CAD/CAM system formation, robot motion planning, NC simulation and mechanism synthesis. In NC verification (Oliver & Goodman, 1990; Jerard et al., 1989; Menon & Volelcker, 1993), the geometry of the machined part can be expressed as the difference between the stock and the swept volume of the cutter. There are two important approaches to describe the swept volume of an arbitrary moving object: one is called Bruce union or solid representation, which is the union of object configurations at all time instances; and another is called the boundary representation of the cutter swept volumes. Envelope theory (Wang & Wang, 1986; Sambandan, 1988) and the Sweep Differential Equation method (Blackmore & Leu, 1992; Blackmore et al., 1996) are examples of the second approach. Based on the sweep differential equation approach, sweep generators, which can calculate and represent the boundary of a swept volume generated by a general moving object undergoing general motion, have been developed (Leu et al., 1995; Wang et al., 1996) and applied to advance existing manufacturing technology (Deng et al., 1996).

Turbine impellers have wide applications in shipbuilding, automotive and aircraft industries. Intensive attention has been paid on designing a high-efficiency profile and actual multi-axis NC machining of a turbine impeller. Since a turbine blade is composed of sculptured surfaces, currently available techniques for multi-axis NC machining of sculptured surface (Takeuchi, 1992; Li & Jerard, 1994; Liu, 1995) may be adopted. It has been proven that five-axis machining of a sculptured surface is one of the most difficult problems for NC programmers due to the complex surface curvature and tool movement. Simulation and verification of the NC program generated manually or automatically becomes essential in machining a turbine impeller.

In this paper, integration of our research on swept volume with the existing CAD/CAM packages is presented. In section two, the **SDE (Sweep Differential Equation)** and **SEDE (Sweep-Envelope Differential Equation)** sweep generators are discussed. In section three, the swept volume approach is applied to NC simulation and verification. In section four, the integrated CAD/CAM system of 5-axis NC machining developed at NJIT is described. An example of machining a turbine impeller is given in section five to demonstrate the approach and the system. Conclusions and remarks are included in section six.

2. SWEEP GENERATORS BASED ON SWEEP DIFFERENTIAL EQUATION APPROACH

A sweep generator which can calculate and represent the boundary of a swept volume generated by a smooth object undergoing an arbitrary smooth motion may be developed based on our research on the sweep differential equation approach to swept volume. The SDE and SEDE methods are included in this research.

2.1 SDE Sweep Generator

Let x be an arbitrary point in n-real space R^n and M a rigid object with boundary surface ∂M. We confine our attention to smooth sweeps of the object σ; they can be represented in the form

$$\sigma_t(x) = \xi(t) + A(t)x$$

where ξ and A are smooth vector-valued and matrix-valued functions defined on the t-interval [0,1] such that ξ(0)=0 and A(0)=I, and all of the A(t) have determinant one and are orthogonal; i.e., $A^T A = I$. The sweep differential equation (SDE) is of the form

$$\dot{x} = X_\sigma(x,t) \equiv \dot{\xi} + \dot{A}A^T(x - \xi)$$

where the dot denotes differentiation with respect to t. X_σ forms the sweep vector field (SVF) of σ.

In the SVF we have a natural decomposition of the boundary of each M(t) of the object: ingress, egress and grazing set (Blackmore et al., 1992). The SDE method is shown to be of advantage in dealing with non-smooth objects or piecewise smooth objects and the SDE sweep module has been applied to our five-axis tool path planing study (Deng et al., 1996).

2.2 SEDE Sweep Generator

The SEDE method starts from calculating the initial grazing points based on the SDE method. The new differential equation is derived by combining the motion equation, the moving object's geometry property and the envelope equation; it has the form

$$\dot{x} = X_\sigma(x,t) + \langle A\partial_x f(\eta), \partial_t X_\sigma(x,t) \rangle |Z|^{-2} Z$$

where $Z = V_\sigma - \langle V_\sigma, N \rangle N$, $V_\sigma = \dot{A}\partial_x f(\eta) - AH(f(\eta))A^T X_\sigma$ and $H(f) = \left[\frac{\partial^2 f}{\partial x_i \partial x_j} \right]$ is the 3×3 Hessian matrix of f. Derivation of above equation is lengthy and is given in another paper of the authors (Blackmore et al., 1996). The SEDE method is shown to have major advantages over (1) Since the dimension of the grazing point set of an n-dimensional object is n-2, the computational complexity is drastically reduced; (2) It provides an automatic connectivity for the computed boundary points that facilitates integration with standard algorithms for visual realization and Boolean operations.

3. APPLICATION OF SEDE APPROACH TO NC SIMULATION

In NC verification, the geometry of the machined part can be expressed as the difference between the stock and the swept volume of the cutter. So a sweep generator, incorporated with a Boolean subtractor and a collision detector, can be used to simulate the material removal process of 5-axis NC milling. In our work, Virtual NC™ (VNC) from Deneb Robotics, Inc. is chosen as the Boolean subtractor and display platform. The SEDE sweep generator described in Section 2.2 is used to replace the convex hull sweep module in VNC for more accurate swept volume representation.

In order to facilitate this integration, we added a step to our SEDE program that translates machine CL data into our characterization of sweeps, and we had to provide oriented output data for our polygonal (triangular) decomposition of the approximate swept volume in the form that is used in the commercial software. The commercial software uses the swept volume data together with a Boolean subtraction algorithm to simulate the material removal on a workpiece subjected to the cutter motion.

An example is given here to show a simulation of a turbine machining using a flat-end tool in a machining environment. Illustrations of the swept volumes and the material removal process produced by the graphical output from the integrated software are shown in Figure 1. Fig. 1(a) shows the simulated machining workcell before machining. Fig.1(b) shows some configurations of the flat-end cutter with diameter d=0.25 and height h=3.0. The grazing points of the cutter swept volume after triangulation and connection are shown in Fig.1(c). Fig.1(d) shows the machined workpiece, which is obtained by performing Boolean subtraction of the cutter swept volumes from the stock.

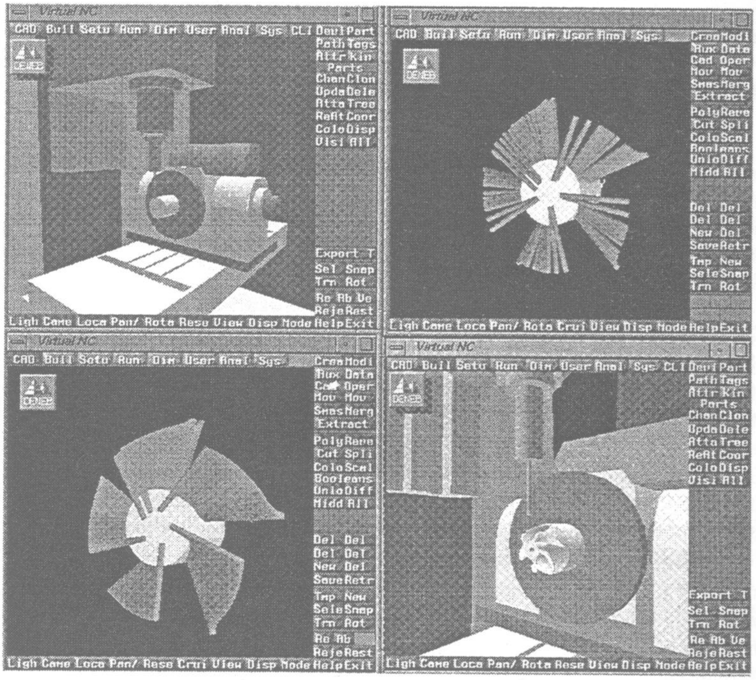


Figure 1. Graphical output of the integration of SEDE sweep module with commercial NC software

4. INTEGRATED 5-AXIS NC MILLING CAD/CAM SYSTEM

The integrated CAD/CAM system of 5-axis NC machining developed at NJIT includes a CAD software for part design, a multi-axis tool path generator, a post processor for transformation of cutter location data to machine control data, a NC simulator/verifier, a five axis NC milling machine and a coordinate measurement machine. A schematic diagram of this system is shown in Figure 2.

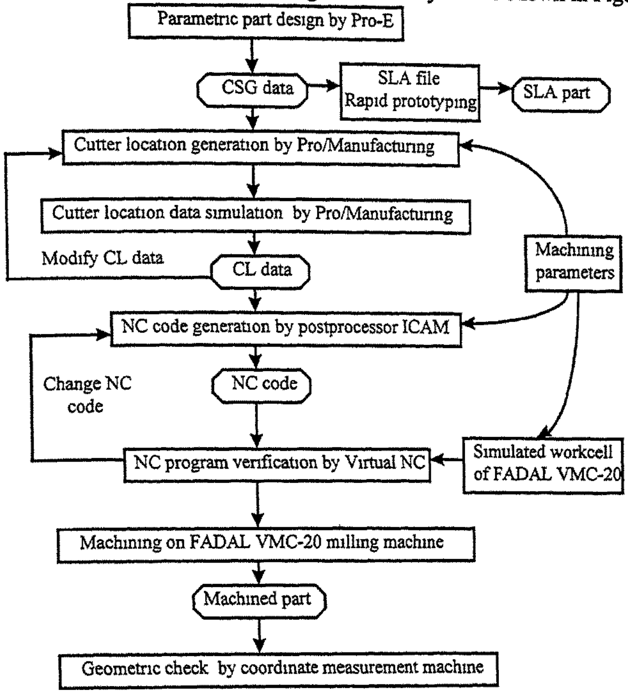


Figure 2. Integrated CAD/CAM system for five-axis NC milling process

CAD software for design: Pro/Engineer™ from Parametric is chosen to module the 3D part. Key features of Pro-E™ include powerful parametric design, unified CSG format, etc.

Rapid prototyping: The designed part by Pro-E™ can be verified by saving the geometry of the part to a "sla" file using the existing function of the software and loading it to 3D system SLA-250 at NJIT to build an actual SLA part using a laser stereolithography technique.

Tool path generation and simulation: Parametric's Pro/Manufacturing™ is used to generate the cutter location data for a given designed part with ordinary features. The boundary of the to-be-machined part, machining parameters and machining strategies have to be determined by the programmer. The same software can be used to simulate the NC milling process using the cutter location data in a stock-tool environment.

NC code generation by post processor: After the generation of the cutter location data, a post processor is needed to transform the CL data into machine axis data. The milling machine we used here is a FADAL VMC-20 vertical milling machine. ICAM™ is incorporated to transform CL data to NC code which is accepted by the controller of the FADAL VMC-20 machine for actual machining.

NC program simulator/verifier: Virtual NC™ from Deneb Robotics, Inc and the SEDE sweep generator developed by the authors are combined to simulate and verify the NC code in a machining environment. To facilitate this process, a workcell was modeled and its kinematics attributes were established using the original machine model of the FADAL VMC-20 machine. By using the Boolean subtractor and verifier in VNC, material removal process is simulated and analyzed in a superior interactive 3D simulation environment. Spatial errors in NC programs (such as gouging and misses) and cycle information (such as cycle times, depth of cut, axis speed, etc.) are detected and measured automatically. Process evaluation and refinement can be performed.

Five-axis machining by FADAL VMC-20: The NC program after the verification by the CAD/CAM system is downloaded to the FADAL VMC-20's controller for actual machining.

Machining error estimate: In this CAD/CAM system, the geometry of the machined part is checked by a coordinate measurement machine and compared with the designed part. The theoretical formula for the error analysis of multi-axis NC milling is also available.

5. CASE STUDY : TURBINE IMPELLER

The turbine blade profile is designed using Pro/Engineer™ based on the theory described in reference (Wilson, 1984) and the part is saved in IGES format for geometric transformation among CAD/CAM software. This part may be saved into a sla file and is sent to 3D system SLA-250 at NJIT for SLA part generation (Fig.3). Verification of the cutter location file and NC program is done using Pro-Manufacturing™, Virtual NC™ and our SEDE approach. The verified NC codes are then loaded in the FADAL VMC-20's controller for actual machining. The machined parts are shown in Fig.3.

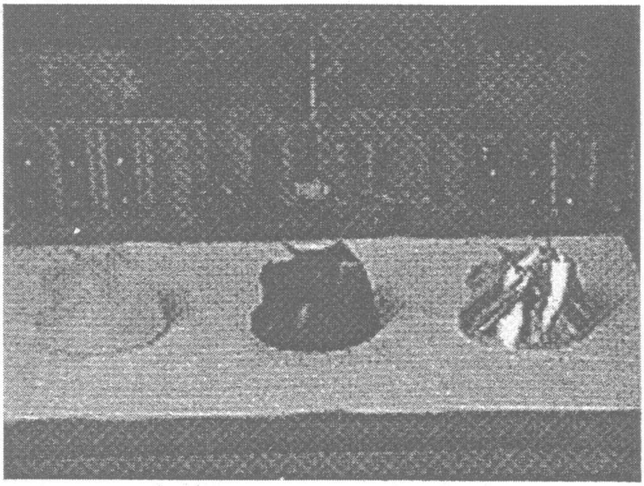


Figure 3. Machined parts: SLA, wax, metal (from left to right)

6. CLOSING REMARKS

The sweep-envelope differential equation (SEDE) method appears to provide a powerful tool for computing swept volumes in space that incorporates both the geometry of the object and the sweep. We have shown that the integration of the SEDE approach with commercial NC software improves the performance by providing a more accurate swept volume representation and error estimates that are especially important in multi-axis NC machining verification.

The integrated CAD/CAM system for five-axis NC milling, which features means for parametric design, several ways of NC program verification in machining environments and five-axis NC milling capacity, has enhanced the design and manufacturing capability of the Center for Manufacturing Systems of NJIT. This system is also assisting several ongoing research projects at NJIT which include (1) application of swept volume in automatic motion planning; (2) tool path generation, error analysis and control of five-axis NC milling; (3) rapid prototyping using laser stereolithography and other projects.

ACKNOWLEDGMENTS

The work of the authors was partially supported by NSF grant DMS-9508808, ONR grant N00014-92-4093 and a Center for Manufacturing Systems grant from NJIT. We would like to thank Dr. Kalin Liu for his help on this project. We would also like to thank the Center for Manufacturing Systems of NJIT for providing the facility for the project development.

REFERENCES

1. Oliver, J.H. and Goodman, E.D., "Direct Dimensional NC Verification", Computer Aided Design, 22(1): pp.3-10, 1990.
2. Jerard, R.B., Hussaini, S.Z., Drysdale, R.L., and Schaudt, B., "Approximate Method for Simulation and Verification of Numerically Controlled Machining Programs", Visual Computer, 5(6), pp.329-348, 1989.
3. Menon, J.P. and Voelcker, H.B., "Toward a Comprehensive Formulation of NC Verification as a Mathematical and Computational Problem", Journal of Design and Manufacturing, 3, pp. 263-277, 1993.
4. Wang, W.P., Wang, K.K., "Real-time Verification of Multi-axis NC Programs with Raster Graphics", Proc. IEEE International Conf. on Robotics and Automation: pp.166-171, 1986.
5. Sambandan, K., "Graphic Simulation and Verification of Five-axis NC Machining", M.S Thesis, School of Mech. and Aero. Engr., Cornell Univ., 1988.
6. Blackmore, D., Leu, M.C., and Wang, K. K., "Applications of Flows and Envelopes to NC Machining", Annals of CIRP, 41/1, pp.493-496, 1992.
7. Blackmore, D., Leu, M.C., and Wang, L.P., "The Sweep-Envelope Differential Equation Algorithm and Its Application to NC Machining Verification", Computer Aided Design, 1996. (accepted).
8. Leu, M.C., Blackmore, D., Wang, L. and Pak, K., "Implementation of SDE Method to Represent Cutter Swept Volumes in 5-Axis NC Milling", Proc. International Conference on Intelligent Manufacturing: pp.211-220, Wuhan, China, 1995.
9. Wang, L.P., Leu, M.C., and Blackmore D., "Generating Swept Solids for General 7-Parameter APT Tools Using SEDE Method", ACM's Symposium on Solid Modeling, Atlanta, GA, 1996. (accepted).
10. Deng, Z., Leu, M.C., Wang, L. and Blackmore, D., "Determination of Flat-End Cutter Orientation in 5-Axis Machining", Proc. ASME International Mechanical Engineering Congress and Exposition, Manufacturing Science and Engineering, MED-Vol.4, pp.73-80, Atlanta, GA, Nov. 17-22, 1996.
11. Takeuchi, Y., and Watanake, T., "Generation of 5-Axis Control Collision-Free Tool Path and Postprocessing for NC Data", Annals of CIRP, 41/1: pp.539-542, 1992.
12. Li, S.X., and Jerard, R.B., "5-Axis Machining of Sculptured Surfaces with a Flat-end Cutter", Computer Aided Design, 26/3: pp.165-178, 1994.
13. Liu, X.W., "Five-axis NC Cylindrical Milling of Sculptured Surfaces", Computer Aided Design, Vol.27, No.12, 1995.
14. Wilson, D.G., The Design of High-Efficiency Turbomachinary and Gas Turbines, MIT Press, 1984.

KINEMATICS ANALYSIS OF GENARAL 5-AXIS NC MILLING MACHINES AND ITS APPLICATION TO NC MACHINING

L.P. Wang, M.C. Leu and *D. Blackmore
Department of Mechanical Engineering
***Department of Mathematics**
New Jersey Institute of Technology
Newark, New Jersey 07102, USA

ABSTRACT

A motion planner for a five-axis NC milling process is developed in this paper. The main advantages of this approach are: (1) Using a generalized kinematics model and a uniform kinematics analysis procedure, it is possible to perform the transformation between cutter location data and machine axis data, thus supporting the NC check in the machining environment without going through a postprocessor; and (2) Integrated with our swept volume generator and a commercial software, it is possible to perform machine axis interpolation to approximate the tool motion between two machine configurations, thus generating more accurate tool swept volume representation for NC machining verification. An example is given to illustrate the method and the results.

KEYWORDS

Kinematics, Multi-Axis NC Machining, Swept Volume

1. INTRODUCTION

For a single step of five-axis NC milling, a typical NC simulator approximates the tool motion between two machine configurations by linear interpolation of the cutter location parameters (Sambandan, 1990), called ***CL interpolation***. It has the form

$$\begin{pmatrix} x_c(t) \\ y_c(t) \\ z_c(t) \\ i(t) \\ j(t) \\ k(t) \end{pmatrix} = \begin{pmatrix} x_c(0) \\ y_c(0) \\ z_c(0) \\ i(0) \\ j(0) \\ k(0) \end{pmatrix} + t \begin{pmatrix} x_c(1)-x_c(0) \\ y_c(1)-y_c(0) \\ z_c(1)-z_c(0) \\ i(1)-i(0) \\ j(1)-j(0) \\ k(1)-k(0) \end{pmatrix} \tag{1}$$

where $x_c(0)$, $y_c(0), z_c(0), i(0), j(0), k(0)$ represent the initial tool configuration and $x_c(1)$, $y_c(1), z_c(1), i(1), j(1), k(1)$ represent the final tool configuration. $t \in [0,1]$ is the normalized time variable. This method does not involve the machine configuration and is convenient in application. However, it causes errors in doing NC check because the actual movement of a NC controller performs a machine axis interpolation instead of cutter location interpolation. Leu et al. (Leu et al., 1995) proposed a different way called ***MCD (Machine Control Data) interpolation*** to describe the motion between any two machine configurations. It has the form

$$\begin{pmatrix} X(t) \\ Y(t) \\ Z(t) \\ A(t) \\ B(t) \end{pmatrix} = \begin{pmatrix} X(0) \\ Y(0) \\ Z(0) \\ A(0) \\ B(0) \end{pmatrix} + t \begin{pmatrix} X(1)-X(0) \\ Y(1)-Y(0) \\ Z(1)-Z(0) \\ A(1)-A(0) \\ B(1)-B(0) \end{pmatrix} \tag{2}$$

where $X(0)$, $Y(0), Z(0), A(0), B(0)$ represent the initial machine configuration and $X(1), Y(1), Z(1)$,

$A(1)$, $B(1)$ represent the final machine configuration. $t \in [0,1]$ is the normalized time variable. But Leu's work did not describe the method in a uniform algorithm and failed to link their approach with an actual NC verification and machine.

The kinematics model for three- to five-axis milling machine has been discussed by Ruegg (Ruegg, 1992). Schwerd's notation was adopted to perform the symbolic kinematics representation of a generalized NC milling machine. Kinematics analysis of five-axis NC milling machines have been proposed to build a real-time five-axis CNC interpolator for machining ruled surfaces (Koren & Lin, 1995). Transformations between CL data and MCD data are derived by geometric relationships. Virtual rotation angles are used to consider different configurations of a five-axis milling machine.

In this paper, the kinematics model of 5-axis NC milling machines is established in a more specific and explicit way. The kinematics analysis of a five 5-axis NC milling machines is reformulated for the purpose of computer-based NC simulation and verification. Transformation between CL data and MCD data is derived using a mathematical method. Integrated with our sweep module and a commercial software (Wang et al., 1997), the NC milling process is simulated in a machining environment and a more accurate geometric representation of the cutter swept volumes is obtained.

This paper is arranged as follows: in section two, a generalized kinematics model for 5-axis milling machines is discussed. In section three, the tasks and procedures of the kinematics analysis of five-axis NC milling model are proposed. Section four presents the kinematics analysis for a FADAL VMC-20 milling machine. An example is given in section five to demonstrate the method. Conclusions and remarks are included in section six.

2. GENERALIZED KINEMATICS MODEL FOR FIVE-AXIS MILLING MACHINES

Considering only the possible arrangements of the rotation axes, five-axis milling machines are characterized into three main families (Leslie, 1970) (illustrated in Fig.1):

Type A: Machines with a rotary table and a tilting spindle head, rotation of the table and head being restricted to one plane;

Type B: Machines with a fixed spindle head and a table capable of rotation in two perpendicular planes;

Type C: Machines with a fixed table and a spindle head capable of rotation in two perpendicular planes.

Certain types of five-axis machines may have different configurations with different tilting and rotation directions. The possible arrangement for the rotation and translation axes, tool and workpiece forms the basis of our generalized kinematics system. The general machine can be specified as a sequence of transformations of coordinates.

3. KINEMATICS ANALYSIS OF FIVE-AXIS MILLING MACHINES

Kinematics analysis of five-axis milling machines is conducted to obtain: (1) the sweep equation or motion equation, and (2) the transformation equations between CL data and MCD data. The kinematics analysis may be done using the geometric properties of the machine (Leu, et al., 1995) or be derived from motion equation mathematically. To perform these tasks, the following procedures are suggested:

Step 1: Find the kinematics parameters of the 5-axis machine in Denavit-Hartenberg notation (Asada & Alotine, 1986). The Denavit-Hartenberg transformation matrix for adjacent coordinate frames, x_i, y_i, z_i and $x_{i-1}, y_{i-1}, z_{i-1}$ is

$$A_i = \begin{pmatrix} \cos\theta_i & -\sin\theta_i \cos\alpha_i & \sin\theta_i \sin\alpha_i & a_i \cos\theta_i \\ \sin\theta_i & \cos\theta_i \cos\alpha_i & -\cos\theta_i \sin\alpha_i & a_i \sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (3)$$

The general machine can be specified as a sequence of transformations of coordinates:

$$T_i = \begin{pmatrix} n_x & s_x & a_x & p_x \\ n_y & s_y & a_y & p_y \\ n_z & s_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{pmatrix} = A_1 A_2 \cdots A_i \quad (4)$$

Step 2: Find the transformation matrices and derive the sweep equation, i.e., the transformation of coordinate systems between tool coordinate frame and workpiece coordinate frame.

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} p_x \\ p_y \\ p_z \end{pmatrix} + \begin{pmatrix} n_x & s_x & a_x \\ n_y & s_y & a_y \\ n_z & s_z & a_z \end{pmatrix} \begin{pmatrix} \xi \\ \eta \\ \zeta \end{pmatrix} \quad (5)$$

where $(x,y,z)^T$ represents any point in the reference coordinate frame and $(\xi,\eta,\zeta)^T$ represents any point in the tool coordinate frame.

Step 3: Perform the inverse kinematics transformation, i.e., transformation of CL data to MCD data. We analyze type B 5-axis milling machines, taking an actual machine (FADAL VMC-20) as an example in the following section. The same procedures may be used for the other types of machines.

4. KINEMATICS PARAMETERS OF FADAL VMC-20

For a Type B machine (FADAL VMC-20) and a fixed configuration as shown in Figure 2, The parameters for the Denavit-Hartenberg notation are as follows:

Link	Frame	α_i	θ_i	a_i	d_i
0	xyz				
1	$x_1y_1z_1$	-90^0	180^0+A	0	$-d_1$
2	$x_2y_2z_2$	90^0	90^0-B	$-d_2$	0
3	$x_3y_3z_3$	90^0	0^0	$-d_3$	X
4	$x_4y_4z_4$	90^0	90^0	0	$Y+d_4$
5	$x_5y_5z_5$	0^0	0^0	0	$Z+d_5$

The transformation matrix A_i for adjacent coordinate frames can be calculated using Eqn.(3).

4.1 Sweep equation

By multiplying the transformation matrices of all links, we obtain the homogeneous transformation matrix T_i as shown in Eqn.(4). Separating T_i into a translation component and a rotation component and using the regular (instead of homogeneous) representation, we obtain the sweep equation:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} p_x \\ p_y \\ p_z \end{pmatrix} + \begin{pmatrix} -\cos B \cos A & -\sin A & -\sin B \cos A \\ -\cos B \sin A & \cos A & -\sin B \sin A \\ \sin B & 0 & -\cos B \end{pmatrix} \begin{pmatrix} \xi \\ \eta \\ \zeta \end{pmatrix} \quad (6)$$

where

$$p_x = X\cos B\cos A + d_2 \cos A \sin B + d_3 \cos A \sin B + h\cos A \sin B \\ -(Z+d_5)\cos A \sin B + d_6 \sin A - (Y+d_4)\sin A,$$

$$p_y = -d_6 \cos A + (d_4 + Y)\cos A - X\cos B \sin A + d_2 \sin B \sin A \\ +d_3 \sin B \sin A + h \sin B \sin A - (d_5 + Z)\sin B \sin A,$$

$$p_z = -d_1 + d_2 \cos B + d_3 \cos B + h\cos B - (Z+d_5)\cos B + X\sin B.$$

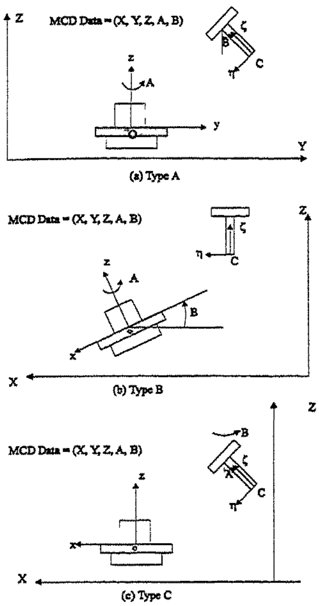


Fig.1 Three types of five-axis NC milling machines

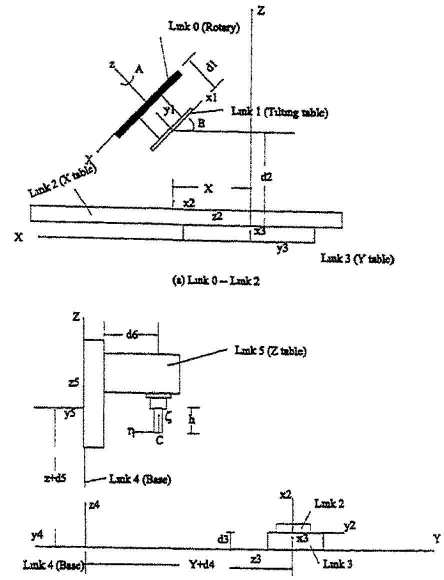


Fig.2 Denavit-Hartenberg representation of FADAL VMC-20 machine

4.2 Transformation of CL Data to MCD Data

The tool axis has direction $(0,0,1)$ in the tool coordinate frame and (I,J,K) in the reference coordinate frame, therefore from Eqn.(6), we obtain the rotational motions

$$A = \tan^{-1}(-\frac{J}{I}) \qquad (0 \le A \le 2\pi) \tag{7}$$

$$B = \tan^{-1}(-\frac{\sqrt{1-K^2}}{K}) \qquad (-\frac{3\pi}{2} \le B \le 0) \tag{8}$$

where I,J,K represent the normalized cosines.

The control point C has coordinates (0,0,0) in tool coordinate frame and (x_c, y_c, z_c) in the reference coordinate frame. So from Eqn.(6), we find that

$$\begin{aligned} X = & -\csc B(d_1 - d_2\cos B - d_3\cos B + d_5\cos B - h\cos B - z_c) - \cot B\{\cos B(d_1 - d_2\cos B \\ & - d_3\cos B + d_5\cos B - h\cos B - z_c) + \sin B\cos A(-x_c + d_2\cos A\sin B + d_3\cos A\sin B \\ & - d_5\cos A\sin B + h\cos A\sin B - d_4\sin A + d_6\sin A) - \sin B\sin A(d_4\cos A - d_6\cos A \\ & - y_c + d_2\sin B\sin A + d_3\sin B\sin A - d_5\sin B\sin A + h\sin B\sin A)\} \end{aligned} \tag{9}$$

$$\begin{aligned} Y = & \sec A(d_4\cos A - d_6\cos A - y_c + d_2\sin B\sin A + d_3\sin B\sin A - d_5\sin B\sin A + h\sin B\sin A) \\ & + \sin B\tan A[\cos B(-d_1 + d_2\cos B + d_3\cos B - d_5\cos B + h\cos B - z_c) - \sin B\cos A(-x_c \\ & + d_2\cos A\sin B + d_3\cos A\sin B - d_5\cos A\sin B + h\cos A\sin B - d_4\sin A + d_6\sin A) \\ & - \sin B\sin A(d_4\cos A - d_6\cos A - y_c + d_2\sin B\sin A + d_3\sin B\sin A - d_5\sin B\sin A \\ & + h\sin B\sin A)] - \tan A[\cos B\csc B(-d_1 + d_2\cos B + d_3\cos B - d_5\cos B + h\cos B - z_c) \\ & + \cot B\cos B(-d_1 + d_2\cos B + d_3\cos B - d_5\cos B + h\cos B - z_c) + \sin B\cos A(-x_c \\ & + d_2\cos A\sin B + d_3\cos A\sin B + d_5\cos A\sin B + h\cos A\sin B - d_4\sin A + d_6\sin A) \end{aligned}$$

$$+\sin B\sin A(d_4\cos A-d_6\cos A+d_2\sin B\sin A$$
$$+d_3\sin B\sin A-d_5\sin B\sin A+h\sin B\sin A)] \quad (10)$$
$$Z=\cos B(-d_1+d_2\cos B+d_3\cos B-d_5\cos B+h\cos B-z_c)+\sin B\cos A(-x_c+d_2\cos A\sin B$$
$$+d_3\cos A\sin B-d_5\cos A\sin B+h\cos A\sin B-d_4\sin A+d_6\sin A)+\sin B\sin A(d_4\cos A$$
$$-d_6\cos A-y_c+d_2\sin B\sin A+d_3\sin B\sin A-d_5\sin B\sin A+h\sin B\sin A) \quad (11)$$

5. IMPLEMENTATION AND EXAMPLE

Based on the model of the FADAL VMC-20 five-axis milling machine and the procedures described in section 3, a kinematics analysis software program has been developed. Inputs of the system are the NC codes to be verified, tool geometry and machine model. The outputs include: (1) the sweep equation, and (2) transformations between CL data and MCD data.

The FADAL VMC-20 machine is modeled in accordance with the actual machine geometry (unit: inch) by a commercial software (illustrated in Fig.3). The kinematics relationship is established based on the machine's kinematics model (Fig.2). The machine's geometric parameters are: d_1= 7.5, d_2=12.1, d_3=2, d_4=13.8, d_5=0, d_6= 13.8. Tool geometry (unit:inch): radius r=0.125, length h=3.0.

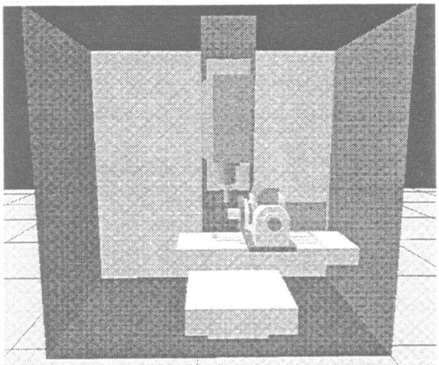


Fig. 3 Simulated workcell of FADAL VMC-20 machine

The NC codes to be verified in this example are part of the codes for machining a turbine. The segment of the codes is :

N237 X-9.8545Y.3014Z.9869A-17.B-0.F516.1
N247 X-9.929Y.3035Z.9254A-13.335F440.9
N254 X-9.9818Y.305Z.884A-10.736F450.4
N265 X-10.0662Y.3072Z.8215A-6.581F464.3
N278 X-10.1682Y.3097Z.7515A-1.556F480.7
N285 X-10.2247Y.3109Z.7152A-358.774F490.
N295 X-10.307Y.3127Z.6651A-354.713F502.7
N305 X-10.3912Y.3144Z.6173A-350.562F514.9
N317 X-10.495Y.3163Z.5629A-345.444F530.7
N329 X-10.6026Z.5116A-340.109F549.
N335 X-10.6579Y.3184Z.4872A-337.356F556.4
...

CL data after the transformation by our kinematics analysis program are:

i	j	k	x_c	y_c	z_c
-0.988568	-0.150771	-0.000587	-1.002230	-0.462105	2.233480
-0.985200	-0.171409	-0.000590	-0.980390	-0.476513	2.252790
-0.972228	-0.234034	-0.000590	-0.887690	-0.525868	2.327330
-0.960616	-0.277878	-0.000587	-0.822770	-0.555524	2.380150
-0.937957	-0.346751	-0.000590	-0.721020	-0.594087	2.464590
-0.903981	-0.427573	-0.000590	-0.601920	-0.627312	2.566630
-0.882163	-0.470944	-0.000587	-0.538200	-0.639767	2.623150
-0.846595	-0.532237	-0.000587	-0.448220	-0.651161	2.705480
-0.805849	-0.592121	-0.000587	-0.360420	-0.654994	2.789710
-0.749814	-0.661649	-0.000587	-0.258550	-0.650004	2.893540
-0.685047	-0.728499	-0.000587	-0.161890	-0.633903	3.001170

and the cutter locations are shown in Fig.4. By using the swept volume generator of our sweep differential equation approach and the Boolean subtractor of a commercial software, the material removal process is simulated and verified as shown in Fig.5.

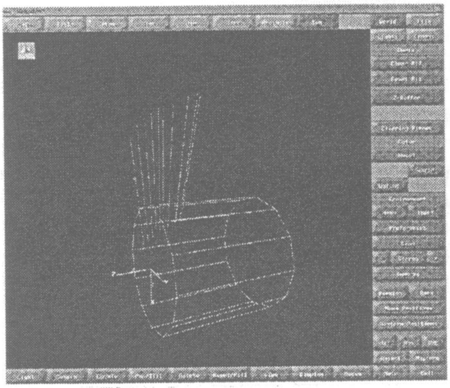


Fig. 4 Cutter locations

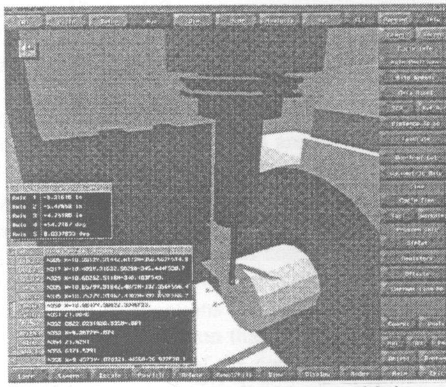


Fig. 5 Simulation of turbine machining

6. CLOSING REMARKS

The generalized kinematics model of 5-axis NC milling machines described in this paper is very effective in conducting the kinematics analysis and building an actual kinematics analysis system. A kinematics analysis system for all three types of five-axis milling machine has been built to generate the tool motion equation and the transformation of CL data to MCD data. Application of the system is addressed to represent cutter swept volume in 5-axis NC milling and verify NC codes. The research has been incorporated with the CAD/CAM system and actual machining at NJIT.

Although the effectiveness of the kinematics analysis system, special attention has to be paid when performing rotational axis transformation of MCD data to CL data, where multiple CL values may exist.

ACKNOWLEDGMENTS

The work was supported by an assistantship from the Dept. of Manufacturing Productivity of NJIT.

REFERENCES

1. Sambandan, K., "Graphic Simulation and Verification of Five-axis NC Machining", M.S. Thesis, School of Mech. and Aero. Engr., Cornell Univ., 1988.
2. Leu, M.C., Blackmore, D., Wang, L. and Pak, K., "Implementation of SDE Method to Represent Cutter Swept Volumes in 5-Axis NC Milling", Proc. International Conference on Intelligent Manufacturing, pp.211-220, Wuhan, China, 1995.
3. Ruegg, A., "A Generalized Kinematics Model for Three-to Five-Axis Milling Machines and Their Implementation in a CNC", Annals of the CIRP, 41/1, pp.547-550, 1992.
4. Koren, Y. and Lin, R.S., "Five-Axis Surface Interpolators", Annals of CIRP, 44/1:379-382, 1995.
5. Wang, L.P., Leu, M.C., and Blackmore D., "Swept Volume Approach as an Integral Part of 5-Axis NC Machining CAD/CAM System", Proc. International Conference on Manufacturing Automation, Hong Kong, 1997. (accepted).
6. Leslie, W.H.P., Numerical Control User's Handbook, Mcgraw-Hill, New York, 1970.
7. Asada, H., and Slotine, J.J.E., Robot Analysis and Control, John Wiley & Sons, pp.31-40, 1986.
8. Fadal Engineering Company Inc., FADAL VMC-20 User's Manual, California, 1994.

COMPUTER AIDED ASSEMBLY PLANNING AND PROCESS MODELLING

J X Gao
School of Industrial and Manufacturing Science
Cranfield University, Building 50,
Cranfield, Bedford, MK43 0AL, UK

and

A Ip
Department of Manufacturing Engineering,
The Hong Kong Polytechnic University
Hong Kong

ABSTRACT

In manufacturing, product assembly accounts for a large proportion of the total cost. This becomes increasingly important as more parts are outsourced. However much less effort has been made in automating assembly planning activities in comparison with machining process. This is mainly because of the complexity of the problem and the lack of scientific research. However, both manufacturing companies and software vendors have shown increasing interest in automatic assembly planning systems. This paper reports on-going research in feature based assembly modelling, knowledge based assembly planning and process modelling for both mechanical and electronics products. The work has been carried out with industrial partners such British Aerospace, PAFEC Ltd and CIM Strategic Research Ltd.

KEYWORDS

Computer Aided Assembly planning, product and process modelling

1. INTRODUCTION

Product assembly is an integrative operation, which is fundamental for the production of nearly all engineering products. It comprises of a series of operations putting together various components to form a final assembly as specified. The detailed sequence and operations are referred to as assembly plans which consist of operation schedules, routing details, machines and tools used and timing details. Assembly planning systems utilise product models and attributes describing the connectivity and relationships between various parts and subassemblies to generate the best assembly sequence. Manual determination of the 'best' assembly sequence for a specific criterion, such as cost is an extremely complicated and difficult task due to the huge number of different assembly sequences that may exist.

Research in computer aided process planning was originally devoted to metal cutting operations for prismatic and cylindrical parts. In the last 30 years tremendous effort has by research centres and industry world-wide [1]. Computer aided assembly planning has not been given sufficient emphasis, especially in the UK. Most work has been done elsewhere. Example are Lin and Chang's (USA) 3-stage assembly plan generation system [2], Delchambre's (Belgium) assembly planning system - GRAFCET [3] and Gu and Yan's (Canada) CAD-directed assembly sequence planning system [4].

Research in assembly planning at Cranfield University has started recently with British Aerospace Defence Dynamics, PAFEC Ltd and CIM Strategic Research Ltd. Some results have been published in 1995 [5,6,7]. A CAPP research team has been established to carry out a comprehensive CAPP requirements capture for BAe and a report has been produced in September 1996 [8]. The work to be reported in this paper includes a feature based assembly modeller developed using a CAD system -

UNIGRAPHICS/PARASOLID, an assembly planning system for mechanical products developed using a manufacturing decision-making system - LOCAM [9] and process modelling for PCB manufacturing.

2. FEATURE BASED ASSEMBLY MODELLING

Currently available solid modellers provide insufficient support for assembly applications. The product modeller developed at Cranfield University can generate a data structure which captures information that is needed for both process and assembly planning, such as geometrical, topological, technological, component and final assembly information. The software tools used are PARASOLID in UNIGRAPHICS environment [10].

Normally the design of an assembly can be presented either by bottom-up approach or by top-down approach, or by a hybrid approach. In the bottom-up approach, individual components and sub-assemblies are designed first and then the assembly relations are defined for them. Whilst, in the top-down approach, design begins with hierarchical decomposition of functions and sub-functions in the form of a system of constraint equations, or an implicit design representation that corresponds to some functionality. The bottom-up approach is used when the detailed designs of the constituent parts of an assembly are available. The top-down approach is better used for conceptual design of a new product where the components are unknown at that stage. The most desirable approach is the hybrid of the two.

The bottom-up approach has been used in this research. The product modeller (figure 1) consists of a user interface, a form feature library, an assembly feature library and a feature editor. PARASOLID was used as the tool to create the assembly model. UNIGRAPHICS was mainly used to display the model. The parasolid data, feature data and assembly data form the complete product model which can be used for downstream applications. It should be noted that the assembly features are used to describe the spatial relationships of discrete parts and can be viewed as a link between two mating features on different parts. Assembly features provide a structure for presenting mutual constraints on shape, dimensions and orientation of mating features.

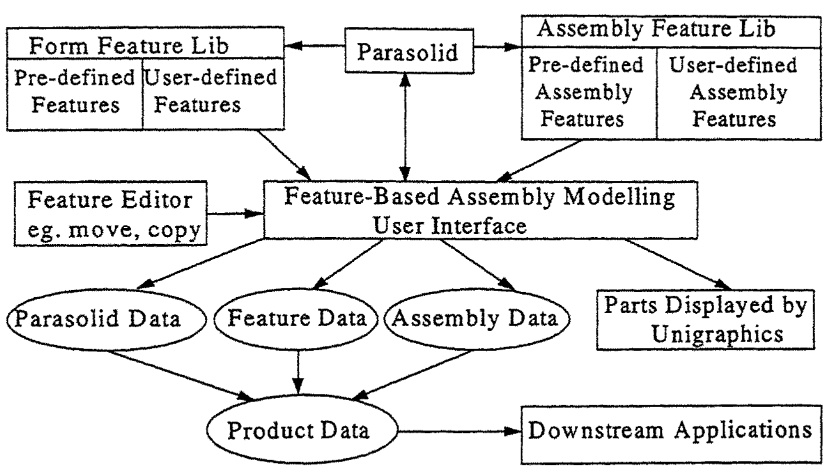


Figure 1: The feature based assembly modelling system

3. KNOWLEDGE BASED ASSEMBLY PLANNING

The assembly planning system developed in this research is a decision-making system which produces assembly plans for a selected family of products. Information about the operations required, the times of handling and insertion operations and the cost on a predetermined operator's cost is provided in the plan. The software tool used is PAFEC LOCAM [9]. LOCAM is a specialised knowledge based system mainly for computer aided process planning purposes. It allows developers and end users to implement interactive, Variant and Generative CAPP systems depending different applications. The simplified planning logic developed in this project is shown in figure 2.

The system begins with a number of questions regarding the assembly name, the operator's name and the operator's cost per hour. Then it proceeds to set up the first component, then its handling and inserting performance data. The information is stored in the "Set first info" block. Next the system asks the user whether any re-orientation or adjustments are required. If the answer is yes, the system sets the re-orientation required and proceeds to the next part. If the answer is no, it skips the set re-orientation block and proceeds to the "next component" block.

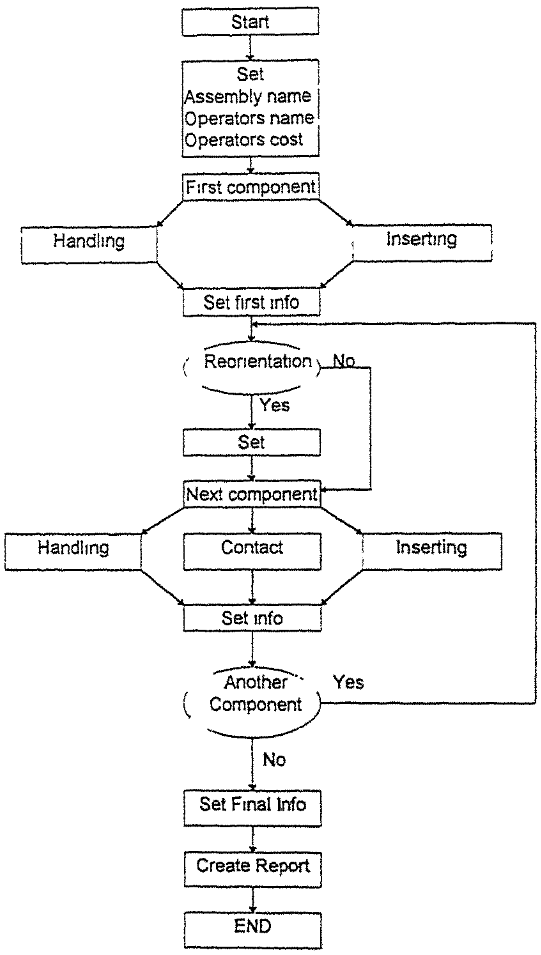


Figure 2: The simplified flowchart of the assembly planning logic

In the "next component" block the name of the component is entered and then the system proceeds to the "handling", "inserting" and "contact" blocks. In these blocks the handling and inserting performance of the block is identified using a number of questions. According to these questions the system estimates the appropriate times required for these operations. In the "contact" block the type of contact is established by the user through a number of predetermined questions regarding contact types. After all the data of the component is entered into the system, it is stored in the memory under the "set info" block. The system then asks whether there is another component of the assembly under the "another component" block. if so, the system returns to the "re-orientation" question block. Otherwise the program sets all the final information regarding the total times and total cost of the assembly process and creates the report. An example of the information entered to the system is given in figure 3, and an example of the planning report is given below to show what information is provided in the report.

Component No: **12**	Component Name: **Plastic cover**
Is the part easy to grasp and manipulate?	YES
Enter the thickness of the part :	51
Enter the size of the part :	155
Enter the alpha symmetry of the part :	360
Enter the beta symmetry of the part :	360
Does the part require holding down to maintain orientation and location ?	YES
Is the part easy to align and position during assembly ?	NO
Is the part secured immediately after assembly ?	NO
Is the part easy to approach during assembly?	YES
Is there any resistance to insertion during assembly ?	NO
Type of Contact required :	SCREW FASTENING
No of screws required :	3
Reorientation Required :	NO

Figure 3: The information required by the planning system

The Assembly Plan Document

Name of the assembly:	Controller assembly	Date:	7 July 1996
Operator's name:	Elias Margellos	Operator cost (£/hr):	20.00

No.	Part name/operation	Handling time	Inserting time	Cost (£)	Description
1	Pressure regulator	2.10	2.50	0.02	place on fixture
2	Metal frame	2.10	5.50	0.04	Add
3	Nut	1.28	14.0	0.08	Add&screw fasten
4	Re-orientation		5.00	0.02	Re-orient&adjust
5	Sensor	2.10	6.50	0.04	Add
6	Strap	1.95	14.50	0.09	Add&screw fasten
7	Adaptor	1.65	15.00	0.09	Add&screw fasten
8	Tube assembly	2.40	12.00	0.08	Add&screw fasten
9	PCB assembly	2.85	14.50	0.09	Add&screw fasten
10	Connector	2.10	9.00	0.06	Insert
11	Earth lead	2.65	9.00	0.06	Add&clip fasten
12	Re-orientation		5.00	0.02	Re-orient&adjust
13	Knob assembly	2.10	13.00	0.08	Add&screw fasten
14	Plastic cover	2.10	18.50	0.11	Add&screw fasten

Total handling time:	25.38 seconds	Total inserting time:	144 seconds
Total time:	169.38 seconds	Total cost:	£0.94 per item

4. PROCESS MODELLING FOR PCB MANUFACTURING

This work has been carried out in collaboration with BAe. The existing process planning system is called PEMS standing for Production Engineering Management System, which was entirely generated in house and went live in the company in 1987 to replace the previous paper hand generated system. Some of the engineering functions are: drawings and sketches, manufacturing instructions, bill of materials, tooling and tool orders, operations scheduling, work flow control and material handling. However PEMS is a variant system which doesn't generate plans automatically. Therefore the company is now using LOCAM to upgrade the process planning system. This research is part of the company's research programme. The first stage is to implement a manufacturing decision system for PCBs - called by BAe as Panel Electronic Circuit (PEC).

There are three stages to the processes involved in PEC manufacturing, i.e., assembly of the electronic components onto the boards (manual or automatic); soldering of the components onto the boards (mass soldering or hand soldering); and rework of any faults from the previous processes and testing of the correct electrical function of the boards. There are also many processes which are included within these three stages, e.g. cleaning and coating the board with protective substance. The objective of this project is to capture the engineer's decision logic in the form of questionnaires and formulate the logic flow of how the engineer decides on which processes, which machines and which other parameters should make up the process plans.

One example of the questions in the logic was "is the board suitable for wave soldering ?" This is of a very complex nature and can only be answered by someone with considerable knowledge of wave soldering. To improve the logic on the system it was required to capture the decision logic used by the process engineers to answer questions such as this. The methodology used is to replace the complex question with a series of geometric queries about the board design, e.g.:

Complex question	**Simpler geometric queries**
"Is the board suitable for wave soldering?"	------------> Ceramic board ?
	|--------> Components with conections on four sides?
	|--------> Surface mounted components?

These simpler queries could make the possibility of CAD and CAPP integration more realistic considering that eventually the geometric questions could be obtained directly from the CAD system. The logic flow to choose route for the company's type B boards is shown in figure 4.

5. CONCLUSIONS

Automating assembly planning is a complex task. It needs more fundamental study of the problem and industrial collaboration and support for research. This paper has introduced the work that is being carried out at Cranfield in product modelling, assembly planning for mechanical and electronic products. The work has been supported by major manufacturing companies and software vendors. This shows that there is a growing interest from industry in this area. Future work includes integrating the three reported tasks through an engineering data management (EDM) or Product data Management (PDM) system. Another project that has been carried out is multimedia information delivery system for CAPP which will be presented at the 32nd International MATADOR Conference to be held in Manchester in July 1997.

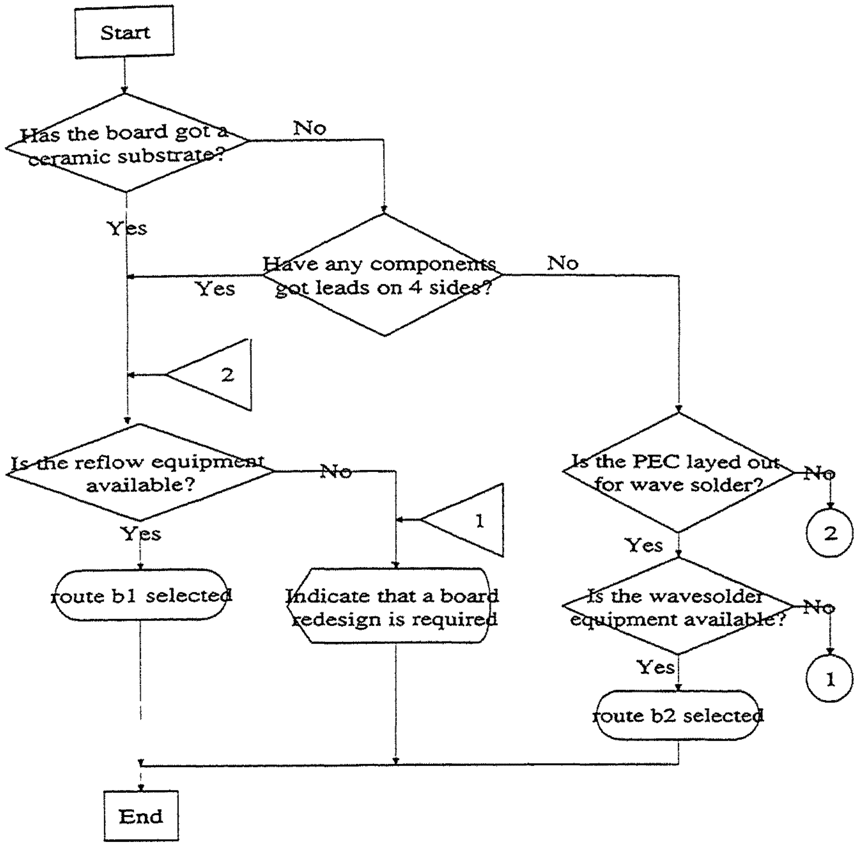


Figure 4: The logic flow to choose the route for type B boards

6. REFERENCES

1. H-C Zhang and L Alting, "Computerized Manufacturing Process Planning Systems", Chapman & Hall, 1994, ISBN 0-412-41300-0
2. A. C. Lin and T. C. Chang, "An integrated approach to automated assembly planning for three-dimensional mechanical products", Int. J. Prod. res., Vol. 31, No. 5, 1993, pp1201-1227.
3. A. Delchambre, Computer-aided assembly planning, Chapman & Hall, 1992.
4. P. Gu and X. Yan, "CAD-directed automatic assembly sequence planning", Int. J. Prod. res., Vol. 33, No. 11, 1995, pp3069-3100.
5. G. R. Bennett and J. X. Gao, "Manufacturing Capability Modelling for Product Design", Integrated Design and Manufacturing in Mechanical Engineering, France, 15-17 April 1996, pp185-194
6. G R Bennett and J X Gao, "The Development of a Rules based CAPP System for Electronic Circuit Assemblies at British Aerospace", The second international conference on Managing Integrated Manufacturing, June 26/27/28 1996, University of Leicester, UK. ISBN 0 9528403 0 8
7. J X Gao, Y S Tang and G R Bennett, "Computer Aided Process Planning for Fabrication and Assembly Applications", 12th National Conference on Manufacturing Research, 9-11 September 1996, University of Bath, UK.
8. Mathrew Swift, Sun Tang and Anna Thorp, "Computer-Aided Process Planning Requirement Capturing Phase I Report", British Aerospace Defence Dynamics, September 1996.
9. PAFEC Ltd., LOCAM User Manual.
10. EDS, PARASOLID and UNIGRAPHICS User Manuals.

A CLUSTERING ALGORITHM FOR AUTOMATED INSPECTION

A.Ajmal & S.G.Zhang*
School of Engineering Systems and Design, South Bank University
103 Borough Road, London SE1 0AA

ABSTRACT

The automated inspection planning for Coordinate Measuring Machines (CMM) should provide a higher accuracy, inspection efficiency with the lower cost. But, the manual inspection planning for a CMM is usually very tedious, time consuming and error prone. An efficient clustering algorithm will enhance the speed and efficiency of the inspection planning process. This entails grouping of the inspection feature into feature families and grouping inspection probes orientations into probe cells. The clustering technique developed will also provide scope for every inspection feature to be inspected at the assigned cell with no or minimal probe calibration and part installation errors. This will result in higher inspection precision, minimal inspection cost with minimal probe change and part re-installation time. This paper outlines the methodology and the main features of development of a knowledge-based clustering algorithm using incidence matrix for the relationship between inspection features and probe orientations. Preliminary tests indicate that the knowledge-based clustering algorithm satisfies the requirement of inspection planning and enhances the speed and accuracy of the inspection process.

KEYWORDS

Clustering algorithm, automated inspection planning, knowledge-based system, probes cell, CMM.

1. INTRODUCTION

Accessibility analysis and probe selection in a CMM system determine probe orientations for each individual inspection feature. However, it is not a good inspection planning method to inspect every individual feature one by one according to the sequence number of inspection feature. The automated inspection planning of CMM should provide a higher inspection accuracy, inspection efficiency and with a lower inspection cost. This requires to group the inspection feature into feature families and to group probe orientations into probe cells. Consequently, there exists no or minimal probe calibration and part installation errors for relative tolerance feature and probe changes and part re-installation time is the least for the entire inspection process. For example, when the perpendicular tolerance is inspected, which requires that the two relevant feature surfaces can be inspected with the same probe at the same installation datum condition. Consequently, there exist no relevant part installation and probe calibration errors for the two relevant inspection features. Moreover, the other feature surfaces that use the same probe can be grouped into the same cluster so that no probe change and part re-installation time are needed when these feature surfaces are inspected. This ensures not only the higher inspection accuracy, but also reduction in inspection time and cost significantly.

A host of algorithms has been published dealing with the formation of cells for group technology technique for manufacturing processes. These aim at obtaining disjoint part families and machine cells such that every part undergoes maximum processing in the assigned cell. Those algorithms include: matrix formulation[1,2,3], mathematical programming formulation[4], graph formulation[5], knowledge-based[6] and neural network-based formulation[7,8].

However, most of these are based on the concept of incidence matrix. For inspection clustering problem, the accessible probe orientations are a continuous angle range that is not represented with incidence matrix directly. Spyridi et al.[9] proposed a clustering method that is a non-empty intersection of accessibility cones. It can guarantee that no collision occurs between probe stylus and part during inspection process. However, the calculation of non-empty intersection is difficult for complex surface.

* To whom correspondence should be addressed.

In this paper, accessibility cone is represented discretely into a set of finite accessibility orientations. That is a conservative subset of accessibility cone. Therefore, it not only guarantees collision free inspection process but also represents the relationship between inspection features and probe cells with incidence matrix.

2. INCIDENCE MATRIX PRESENTATIONS OF INSPECTION TASK

Accessibility cone is represented discretely into a set of finite accessibility orientations. For example, the accessibility cone of a plane is a halfspace as shown at Figure 1a). It can be represented into a set of the coordinate axis direction such as X+, X-, Y+, Y-, Z+ as shown at Figure 1b).

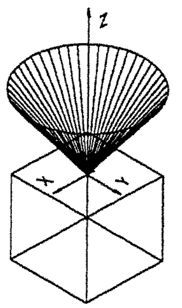


Figure 1a). Accessibility cone of a plane.

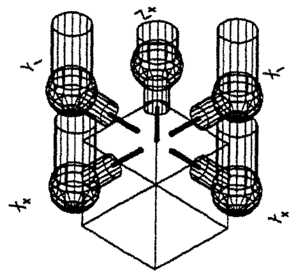


Figure 1b). The discrete probe orientations.

The relationship between the tolerance features and their probe orientations can be represented with incidence matrix. Table 1 shows the incidence matrix between tolerance features and probes orientations. The column of the incidence matrix presents the tolerance features. The tolerance features are listed in the order of 1,2,3,4,5,6,7 and 8. These could be simple dimensional tolerance such as diameter, distance, or the single geometrical tolerance such as flatness, cylindricity, or the related geometrical feature such as parallelism, perpendicular, etc. The row of the incidence matrix presents the probe orientations. The probe orientations are represented as X+, X-, Y+, Y-, Z+, Z-. If the inspection of the *j*th tolerance feature uses the *i*th probe orientation, the element a_{ij} of the incidence matrix is set to be equal to 1, and otherwise, is set to be 0. For single tolerance feature, only a probe orientation is needed to implement the inspection. For complex or related tolerance feature, therefore, a few of probe orientations are needed to implement the inspection. The creation of incidence matrix should consider not only the tolerance relationship between inspection features, but also probe orientation selection from the set of accessibility probe orientations. This is implemented with a knowledge-based clustering algorithm. The details are beyond the scope of the paper and that will be discussed in a separate paper later.

Probe orientation	Tolerance Feature							
	1	2	3	4	5	6	7	8
X+		1	1		1			
X-	1					1		
Y+				1			1	
Y-	1					1		
Z+				1				
Z-		1	1		1			1

Table 1. Incidence matrix of inspection features and their probe orientations.

3. CLUSTERING ALGORITHM

Cluster analysis is concerned with grouping of objects into homogeneous clusters based on the object feature. The application of cluster analysis in probe selection is to group features into feature families and probes into probe cells such that every inspection feature can be inspected at the assigned probe cell. In the cell, the parts' installation datum is not changed and the probes are not changed yet or changed a very little. This will guarantees that high measurement precision and will reduce inspection cost significantly.

The cluster identification algorithm allows one to check the existence of mutually separable in a binary machine-part incidence matrix, provided that they exist. For inspection task, the incidence matrix consists of related factors a_{ij} between probe and feature. If the *i*th probe can be used to inspect the *j*th feature, then $a_{ij} = 1$, otherwise $a_{ij} = 0$.

The procedure for clustering algorithm is as follows:

- Step 0. Set iteration number $k = 1$ (group number of cluster).
- Step 1. Select a row i of the incidence matrix $A^{(k+1)}$ and draw horizontal line h_i through the row i.
- Step 2. Draw a vertical line v_j for each entry 1 crossed by horizontal line h_i.
- Step 3. Draw a horizontal line h_k for each entry 1 crossed by vertical line v_j.
- Step 4. Repeat steps 2 and 3 until there are no more crossed-once entries of 1 in $A^{(k+1)}$. All crossed-twice entries of 1 in $A^{(k+1)}$ form probe cell PC-k and feature family FF-k.
- Step 5. Delete the rows and columns that are drawn by steps 1 to 4. A new incidence matrix $A^{(k+1)}$ is obtained.
- Step 6. If $A^{(k+1)} = 0$ (where 0 denotes a matrix with all elements equal to zero), then stop, otherwise set $k = k+1$ and go to step 1.

Table 2 shows the result of clustering algorithm for Table 1. That is a block diagonal matrix. It can be seen from Table 2 that tolerance features 2,3,5 and 8 (feature family 1) are only related with probe X+ and Z- (probe cell 1); tolerance features 1 and 6 (feature family 2) related with probes X- and Y- (probe cell 2); and tolerance features 4 and 7 (feature family 3) related with probes Y+ and Z+ (probe cell 3).

Probe orientation	Tolerance Feature							
	2	3	5	8	1	6	4	7
X+	1	1	1					
Z-	1	1	1	1				
X-					1	1		
Y-					1	1		
Y+							1	1
Z+							1	

Table 2. Incidence matrix after clustering.

The clustering algorithm produces the mutually separable (as in Table 2) or partially separable cluster (as in Table 4 and Table 5). In the same group, there exists tolerance relationship between inspection features or the same probes are used for the different features. Within the cluster, the installation datum of part to be inspected is not changed and probes orientations are not changed yet if possible. This provides the same installation datum and calibration error. Therefore, no relevant installation and calibration errors. This guarantees that high inspection precision can be obtained and the probes to be used are as few as possible. Moreover, the probe calibration time and cost are reduced significantly.

4. KNOWLEDGE-BASED PROBE SELECTION (KBPS) SYSTEM

In this section, a knowledge-based probe selection clustering algorithm system (KBPS) is discussed. The KBPS has the structure as shown in Figure 2.

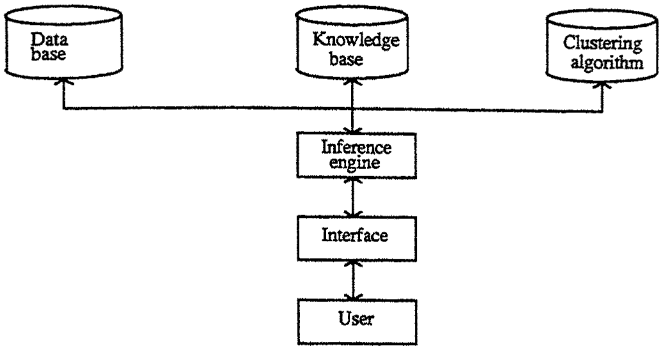


Figure 2. Knowledge-based clustering algorithm system.

First, KBPS constructs a probe-feature incidence matrix based on the data provided by the user before the grouping process. Next, the KBPS initialises the data base. All the initial parameters and matrix are assigned. Then the system forms probe cells and corresponding inspection feature families. Each probe cell is formed by including only one probe at a time. Each time a probe cell has been formed, the KBPS checks whether the constraints have been violated and removed all features violating the constraints. For a probe cell that has been formed and analysed by the KBPS relative to the constraints, the corresponding features forming a inspection feature family are removed from the probe-feature incidence matrix. The process is repeated until the probe-feature incidence matrix become zero.

5. EXAMPLE OF PROBE SELECTION

The proposed clustering algorithm is tested with an example part as shown at Figure 3. That comprises 6 planes F1, F2, F3, F4, F5, F6 and 3 through-holes F7, F8, F9.

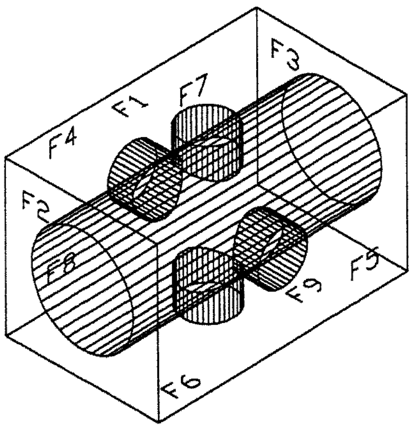


Figure 3. Part to be inspected.

The example part includes 12 tolerance feature 1, 2, ..., 12. They are distance, diameter, perpendicular, flatness, cylindricity, parallelism, etc. dimensional and geometrical tolerances. The incidence matrix between tolerance features and probe orientations is shown at Table 3.

Probe orientation	Tolerance Feature											
	1	2	3	4	5	6	7	8	9	10	11	12
X+			1			1			1	1		
X-			1	1	1	1			1	1		
Y+	1						1		1		1	
Y-							1	1	1			
Z+			1									1
Z-		1	1									

Table 3. Incidence matrix between tolerance feature and probe orientations.

For the different constraints the result of clustering algorithm is different. For example, when the maximum probe number in a cell is 5, the result of clustering algorithm is shown at Table 4. It can be seen that the probe cell 1 comprises probe orientations X+, X-, Z+, Z-, and feature family 1 comprises feature 3, 4, 5, 6, 10, 12 and 2. The probe cell 2 consist of Y+, Y-, and feature family 2 consist of feature 1, 7, 8 and 11. The feature 9 is put at waiting list since it uses the probes that are used in cell 1 and cell 2 simultaneously.

Probe orientation	Tolerance Feature											
	3	4	5	6	10	12	2	1	7	8	11	9
X+	1			1	1							1
X-	1	1	1	1	1							1
Z+	1					1						
Z-	1						1					
Y+								1	1		1	1
Y-									1	1		1

Table 4. Clustering algorithm results when maximum probe number is 5.

When the maximum probe number in a cell is 2 the result of clustering algorithm is shown at Table 5. The probe cell 1 include probe orientations X+, X-, and feature family 1 include feature 4, 5, 6 and 10. The probe cell 2 consist of Y+, Y-, and feature family 2 consist of feature 1, 7, 8 and 11. The probe cell 3 comprises Z+, and feature family 3 comprises feature 12. The probe cell 4 comprises Z-, and feature family 4 comprises feature 2. The feature 3 and 9 is put at waiting list since they use the probes that belong to more than two cells. The features that are put at waiting list should be inspected individually.

Probe orientation	Tolerance Feature											
	4	5	6	10	1	7	11	8	12	2	3	9
X+			1	1							1	1
X-	1	1	1	1							1	1
Y+					1	1	1					1
Y-						1		1				1
Z+									1		1	
Z-										1	1	

Table 5. Clustering algorithm results when maximum probe number is 2.

The application of a knowledge-based clustering algorithm provides opportunity for inclusion of more constraints and can also generate different cluster search direction for different inspection problems. Each time, a probe cell has been formed, the knowledge-based probe selection clustering algorithm system (KBPS) checks whether the constraints have been violated and removed all features violating the constraints. For a probe cell that has been formed and analysed by the KBPS, the corresponding features forming a feature family are removed from the probe-feature incidence matrix.

6. CONCLUSIONS

In this paper, the relationship between inspection features and their probe orientations is represented with an incidence matrix. The concept of probe selection is similar to clustering algorithm in group technology and a clustering algorithm for inspection planning process is feasible. A mutually separable or partially separable cluster is obtained with the proposed clustering algorithm. In the same cluster, there exists either relative tolerance relationship between inspection features or using the same probe for the different features. Grouping relative tolerance feature into the same cluster that guarantees no relevant installation and calibration error for the relative tolerance feature. Grouping inspection features into the same cluster in which the same probe is used, consequently, that provide the fewest probe changes and part installation time. These ensure that a higher inspection accuracy and lower inspection cost can be obtained. Test results proves that the proposed clustering algorithm satisfies the requirement of inspection feature grouping.

Unfortunately, the minimal clustering problem is NP-complete. No analysis solution exists for this problem. Minimal clustering does not generally lead to unique solution. In CMM inspection problems the quality of several solutions may be different, and may depend on complicated considerations about workpiece stability, setup cost, and fixturing. Ideally, such factors should be taken into account in the clustering algorithm, to generate the "best" solution among those with minimal number of probes. In current work, a knowledge-base clustering algorithm is used that results in more factors can be considered with constraints. However, an approximation optimum solution can be obtained.

7. REFERENCES

1. Srinivasan, G., Cell formation using zero-one matrices — experiences with real-life data and inferences, Computers & Industrial Engineering, 29, 1-4, p303-307, 1995.
2. Levasseur, G.A., Applications-oriented procedure for cell formation, Production and Inventory Management Journal, 37, 1, p7-11, 1996.
3. Perrego, T.A., Petersen, H.C. & Hahn, W.F., Perrego algorithm: a flexible machine-component grouping algorithm based on group technology techniques, International Journal of Production Research, 33, 6, p1709-721, 1995.
4. Lee, H. & GarciaDiaz, A Network flow approach to solve clustering problems in group technology, International Journal of Production Research, 31, 3, p603-612, 1993.
5. Rath, S.K., Das, C. & Sahu, S., Graph colouring approach for cell formation problems, Production Planning and Control, 6, 5, p421-428, 1995.
6. Kusiak, A., Intelligent manufacturing system, Printice-Hall International Inc., 1990.
7. Kamal, S. & Burke, L.I., FACT: a new neural network-based clustering algorithm for group technology, International Journal of Production Research, 34, 4, p919-946, 1996.
8. Kaparthi, S., Suresh, N.C. & Cerveny, R.P., Improved neural network leader algorithm for part-machine grouping in group technology, European Journal of Operational Research, 69, 3, p342-356, 1993.
9. Spyridi, A.J. & Requicha, A. A. G., Accessibility analysis for the automatic inspection of mechanical parts by coordinate measuring machines, Proceedings of IEEE International Conference on Robotics and Automation, Cincinnati OH, p1284-1289, 1990.

A TOOLPATH AND CUTTING DEPTH ALGORITHM FOR ROUGH MACHINING

Y. N. Hu, W. C. Tse, Y. H. Chen
Department of Mechanical Engineering
The University of Hong Kong
Hong Kong

ABSTRACT

In this paper, the commonly-used toolpath patterns are summarized A knowledge-based parametric approach for optimizing the toolpath pattern of a given cutting layer is presented Then, a novel methodology is developed to calculate an arbitrary polygon area and locate the concave cavities on the polygon. The procedures for cutting layer shape analysis and the optimal comprehensive toolpath pattern generation are also built and proposed in this paper These procedures could not only be applied to sculptured cavity parts with simple islands but also to parts with arbitrarily-shaped islands. Finally, an example is given to illustrate the reasoning process.

KEYWORDS

NC Toolpath Generation, Surface Machining, Process Planning, Rough Machining

1. INTRODUCTION

Rough machining takes the majority of machining time when a sculptured part is machined from a prismatic stock Traditionally, frequently-used rough machining method for molds and dies is to cut part surface layer by layer using the contour-map approach, as illustrated in Figure1 Related problems such as efficient cutter motion selection, workpiece orientation optimization, and more frequently, efficient toolpath pattern planning therefore have been widely studied by many researchers.

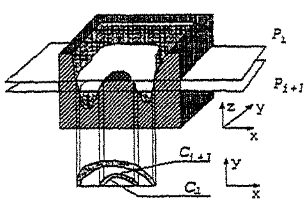


Figure 1. Illustration of Contour Map Approach

For a sculpture part shown in Figure 1, according to the variety and complexity of the geometry of the stock and the part, cutting layer shapes can be classified into three groups: layer shapes with no island, a single island and multiple islands. To machine a sculpture surface part layer by layer, so far, there are six frequently used toolpath patterns in contour-map machining[1]. These toolpath patterns include *Stock-offset pattern (SO), Component-offset pattern (CO), Stock/component-offset pattern (SCO), Parallel-offset pattern (PO), Proportional-blending-offset pattern (PBO), and Max-min-offset pattern (MMO).* Each pattern has its most efficient application case based on the advantages and disadvantages of six toolpath patterns. It is noted that the frequency of plunge, retract and traversing motions is the major factor that influence the efficiency of a toolpath pattern. This paper provides a comprehensive study of toolpath patterns and optimal cutting depth calculation.

When analyzing whether or not a particular toolpath pattern is suitable for the given part, productivity of such toolpath patterns should be evaluated. The total production time, T, is dominated by two terms of Equation (1) as follows, the cutting time T_c and the tool rapid traverse time T_m. The total production time of rough machining, is formulated as follows[2]:

$$T = \sum_{i=1}^{N} t_i = \sum_{i=1}^{N} t_{ci} + \sum_{i=1}^{N} t_{mi} \qquad (1)$$

Based on Equation (1), when the feedrate and the speed of rapid traverse are constant, the production time T is only determined by the number of cutting layers N and the length of toolpath and rapid

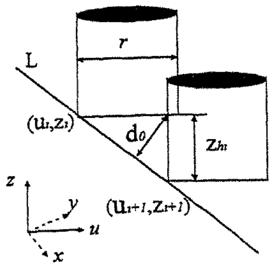


Figure 2 Cutting Depth Planning

traverse trajectory on each cutting layer. Therefore, the cutting depth on each layer also needs to be appropriately chosen in order to obtain high productivity.

The method of determining the cutting depth is outlined below: Suppose the machining accuracy (cusp height d_0) for a part is given, the cutting depth of each cutting layer can be determined by an iterative approach Consider a vertical section that is formed by cutting the part by a plane parallel to the z axis and passing a CC point as shown in Figure 2, the equation of the straight line segment used to approximate the part profile is found to be $Az + Bu + C = 0$, where $u \in \{ax + by + c\}$ and the intersection point between the tool profile of the previous cutting layer and the tool profile of this layer is (u_{i+1}, z_i). Then the cutting depth of this layer Z_h can be determined by solving equations $|Az_i + Bu_{i+1} + C| \le d_0(A^2 + B^2)^{1/2}$ and $u_{i+1} = (x_{i+1}^2 + y_{i+1}^2)^{1/2}$. The above procedure is repeated for each CC points of this cutting layer and the minimum one found is taken as the actual cutting depth Z_h for this layer. A detail description of this method can be found in the paper[3].

2. CRITERIA FOR ANALYZING STOCK AND ISLAND SHAPES

We suppose that on a particular cutting layer, any region bounded by boundary is called a component. The contour of island component i consists of ml edges and its circumference is $L_i = \sum_{k=1}^{ml} l_k$ where l_k is the length of kth edge, as shown in Figure 3(a). The circumference of cavity contour is L_0. Moreover, A_i and A_0 respectively denote area of ith island and area that is bounded by L_0. α_i and β_i specify the centroid for ith component, as shown in Figure 3(b).

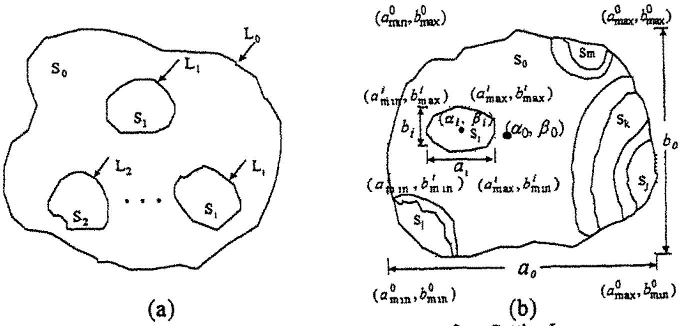


Figure 3. Definition of the Geometric Component for a Cutting Layer

So far in the system, the parameters used to describe the features of cutting layer shape include:

(1) Island Number (s_1): It is the number of islands on each cutting layer.

(2) Island/Part Area Ratio (s_2): It is the ratio of the island area to the machining area bounded by the outermost boundary contour and is an identification parameter for machining feasibility. s_2 will be discretilized into 4 types: 'small', 'medium', 'large' and 'huge'

(3) Shape Convexity-Concavity (s_3): It is an identification index for machining feasibility. The part shapes are first analyzed by the procedure shown in next section. Then toolpath patterns are planned.

(4) Shape Length/Width Ratio (s_4): The extreme points of the polygon are defined by $a^i_{min} = \min\{x_k\}_{k=1}^{mi}$, $a^i_{max} = \max\{x_k\}_{k=1}^{mi}$, $b^i_{min} = \min\{y_k\}_{k=1}^{mi}$, and $b^i_{max} = \max\{y_k\}_{k=1}^{mi}$, where the point (x_k, y_k) denotes the vertex coordinate of the polygon of the ith island or the polygon of outermost boundary and mi is the number of polygon vertices (Figure 2(b)). After knowing the extreme points, a rectangular box with vertices (a^i_{min}, b^i_{min}), (a^i_{max}, b^i_{min}), (a^i_{max}, b^i_{max}) and (a^i_{min}, b^i_{max}), is obtained. Its edges have the lengths $a_i = (a^i_{max} - a^i_{min})$ and $b_i = (b^i_{max} - b^i_{min})$. So s_4 could be calculated and

determined by a_i/b_i. s_4 also needs to be discretilized into three different types, namely 'large', 'medium', and 'small'. If the shape is long and thin (i.e. s_4>>1 or s_4<<1), it is set as 'large'. On the other hand, if the island shape is short and thick (i.e. $s_4 \approx 1$), it is set as 'small'. If the shape is in the between of these two extremes, it is set as 'medium'. Generally s_4 can be used to justify what kind of tool path pattern is suitable for a particular shape.

(5) Island Eccentricity (s_5) : This parameter shows the postional relation between the island centroid and the cavity boundary centroid (see Figure 1(b)). The *i*th island shape centroid and boundary shape centroid could be roughly calculated by $\alpha_i = \bar{x} = (\sum_{k=1}^{mi} x_k)/mi$ and $\beta_i = \bar{y} = (\sum_{k=1}^{mi} y_k)/mi$, in which the point (x_k, y_k) and *mi* have the same definition as above. s_5 is only significant in the case of s_2 ='small' or 'medium', and $s_4 \approx 1$. Eccentricity values along both x and y direction could be calculated separately as follows:

$$s_{5x} = ((\alpha_0 - \alpha_i)^2 + (\beta_0 - \beta_i)^2)^{1/2} / a_i$$

$$s_{5y} = ((\alpha_0 - \alpha_i)^2 + (\beta_0 - \beta_i)^2)^{1/2} / b_i$$

where (α_0, β_0) and (α_i, β_i) represent the cavity shape centroid and island shape centroid respectively. s_{5x} and s_{5y} denote the eccentricity degree along x and y directions respectively. In our system, s_5 also needs to be discretilized into three different types, namely 'large', 'medium', and 'small'. It can also be used to determine tool path pattern together with other parameters.

(6) Machining Area Relation (s_6): It is used to arrange the operation sequence for the given machining areas when cutting a layer. If the machining areas are close together, for example for the areas S_k and S_j shown in Figure 1(b), they will be assigned a relation 'link' between them. Moreover, for the situation of Figure 1(b), since the operation order $S_j \rightarrow S_k \rightarrow S_0 \rightarrow S_l \rightarrow S_m$ would have relatively few traversing movements, the areas S_k and S_j will therefore be set to 'Link' relation and areas S_j and S_l , and S_j and S_m will be set to 'Non-link' relation. s_6 is a list-structure parameter that indicate the machining sequence for the given machining areas on a layer.

(7) Pattern Planning Number (s_7) : It is used to tell how many times of toolpath pattern planning is needed to generate for the given cutting layer.

(8) Component/Box Area Ratio (s_8): It is calculated by $A_i/(a_i b_i)$, where A_i denotes the *i*th component area, and a_i and b_i are the length of the edges of the component box as defined in the parameter s_4. For *i=0*, it denotes the area within the cavity of the part. s_8 usually needs to be used together with the parameters s_3 and s_9 for use.

(9) Component/Box Circumference Ratio (s_9): It is calculated by $L_i/2(a_i+b_i)$, where L_i is the component contour circumference on the cutting layer, and a_i and b_i are the length of the edges of the component box as defined in the parameter s_4.

3. CONTOUR SHAPE CLASSIFICATION AND OPTIMAL TOOL PATH PATTERN GENERATION

3.1 Formula for Calculating the Arbitrary Polygon Area

It could be proved that for an arbitrary polygon with *m* vertex points $\{Q_1, Q_2, \cdots Q_m\}$, its area S is able to be calculated by the following formula:

$$S = 0.5 \sum_{i=1}^{m-2} (Q_{i+1} - Q_1) \times (Q_{i+2} - Q_1) \qquad (2)$$

3.2 Procedure to Determine the Location of Shape's Concave Cavity (LSCC)

The shape's concave cavity often appears with the form of compound cavity. Compound cavity is a cavity with several small cavities and protrusions inside. For example, in Figure 4 the shape of the polygon between the vertices Q_6 and Q_{13} ought to be defined as a compound cavity because though the protrusions exist at points Q_9 and Q_{11}, they couldn't change the overall concave shape of the cavity. To determine the location of compound cavity (for example, in Figure 4 the start point p_s and the end point p_e should be determined), the relation between sign of area term and parameters for judging the convexity-concavity mentioned above need to be used, and the procedure are as follows.

Step 1: Calculate each term of Equation (2) and build arrays $\{S_1^1, S_2^1, \cdots, S_{m-2}^1\} \ldots \{S_1^m, S_2^m, \cdots, S_{m-2}^m\}$, where each term value could be obtained by following equations:

$$S_1^1 = 0.5\{(\mathbf{Q}_2^1 - \mathbf{Q}_1^1) \times (\mathbf{Q}_3^1 - \mathbf{Q}_1^1)\} \ldots S_{m-2}^1 = 0.5\{(\mathbf{Q}_{m-1}^1 - \mathbf{Q}_1^1) \times (\mathbf{Q}_m^1 - \mathbf{Q}_1^1)\}$$

$$\vdots$$

$$S_1^m = 0.5\{(\mathbf{Q}_2^m - \mathbf{Q}_1^m) \times (\mathbf{Q}_3^m - \mathbf{Q}_1^m)\} = 0.5\{(\mathbf{Q}_1^1 - \mathbf{Q}_m^1) \times (\mathbf{Q}_2^1 - \mathbf{Q}_m^1)\}$$

$$\vdots$$

$$S_{m-2}^m = 0.5\{(\mathbf{Q}_{m-1}^m - \mathbf{Q}_1^m) \times (\mathbf{Q}_m^m - \mathbf{Q}_1^m)\} = 0.5\{(\mathbf{Q}_{m-2}^1 - \mathbf{Q}_m^1) \times (\mathbf{Q}_{m-1}^1 - \mathbf{Q}_m^1)\}$$

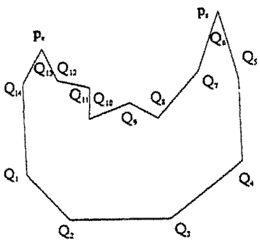


Figure 4. Illustration of Concave Cavity

In the above equations, $\mathbf{Q}_1^1$ is the original vertex, and when taking such original vertex as start point, $\mathbf{Q}_i^1$ means the *i*th vertex point by counting the vertices along the counterclockwise direction from $\mathbf{Q}_1^1$. Similarly, if $\mathbf{Q}_1^j$ takes the *j*th vertex as start point referring to original point $\mathbf{Q}_1^1$, then $\mathbf{Q}_i^j$ means the *i*th vertex point referring to the start point $\mathbf{Q}_1^j$ by counting the vertices along the counterclockwise direction. In other words, the point $\mathbf{Q}_i^j$ actually corresponds to the point $\mathbf{Q}_{\text{mod}((i+j)/m)}^1$ which is the *mod* ((*i*+*j*)/*m*) vertex counting from $\mathbf{Q}_1^1$, where the function *mod* ((*i*+*j*)/*m*) means to take the residue value of ((*i*+*j*)/*m*).

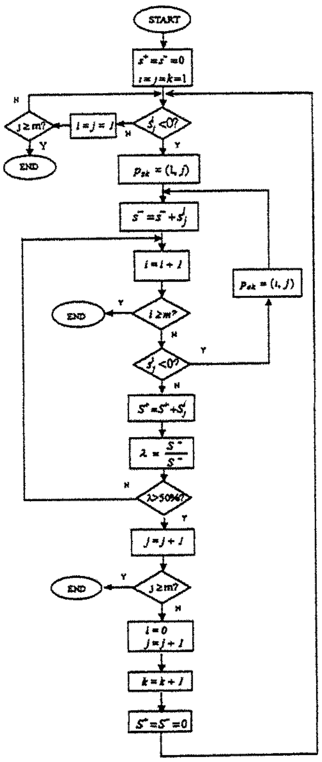


Figure 5. Flow Chart for Finding Cavities

Step 2: Determine the location of concave cavity using the process illustrated in Fig. 5. Since a compound concave cavity may have protrusions inside, and if such protrusions are not big enough, it should be considered as a single cavity, so the process shown in Fig. 5 should be built and adopted to check the location of compound concave cavity. Here, if the ratio of the sum of positive area elements to the sum of negative area elements is smaller than 50%, it will be classified as a compound concave cavity. Otherwise, the adjacent cavities will be viewed as different compound cavities.

Step 3: List all of the compound concave cavities and store their start and end vertices $\{p_{sk}, p_{ek}\}|_{k=1}^{l}$ into the system's database, where p_{sk} and p_{ek} are the start and end vertices of *k*th compound concave cavity and l is the number of total detected compound concave cavities.

3.3 Procedure of Optimal Toolpath Pattern Generation(OTPG)

The parameters that describe the shape of polygon without island are s_3, s_4, s_8 and s_9. So, to classify the shape of arbitrary polygon, we should first calculate these parameters.

Step 1: Compute the given polygon area using Equation (2). Apply LSCC procedure to find out all of the locations of compound concave cavity $\{p_{sk}, p_{ek}\}|_{k=1}^{l}$.

Step 2: Calculate s_8 and s_9.

Step 3: Calculate s_4.

Step 4: By the result of LSCC procedure and

Equation (2), find out all negative areas of concave cavities on the given polygon. Then calculate the index as below:

$$\kappa_k = \frac{|S_k|}{S} \quad k=1,..,l \qquad (3)$$

where S is the total polygon area, S_k is the area of kth compound concave cavity, and κ_k represents the ratio of the concave cavity area to the total polygon area.

To determine parameter s_3, as shown in Figure 6, the shape classification for arbitrary polygons with no island could be classified into eight type shapes: CS, NS, SCS, MCS, LMC, LCS, LLC, and IS.

For arbitrary polygons stated above, only four toolpath patterns, SO pattern, PO pattern, MMO pattern and PBO pattern, are suitable. Frequently, these four toolpath patterns need to be comprehensively adopted together to generate the toolpath for a particular part. For simplicity, the following only gives a brief description for the application of OTPG process for an arbitrary polygon without island inside.

(1) If the given polygon is CS shape, firstly s_4 is computed. If $s_4 \approx 1$, then SO and PO patterns are candidate patterns, in which the pattern with shorter machining time is finally selected as the toolpath pattern. Otherwise, if $s_4 >> 1$ or $s_4 << 1$, then SO pattern is chosen as the optimal toolpath pattern.

(2) If the given polygon is NS shape, then SO and PO patterns are candidate patterns, in which the pattern with shorter machining time is finally selected as the optimal toolpath pattern.

(3) If the given polygon is SCS shape, SO pattern is selected as the optimal toolpath pattern.

(4) If the given polygon is MCS shape, SO and PBO patterns are candidate patterns, in which the pattern with shorter machining time is finally chosen as the optimal toolpath pattern.

(5) If the given polygon is LMC shape, SO, MMO and PBO patterns are candidate patterns, in which the pattern with the shortest machining time is finally selected as the optimal toolpath pattern.

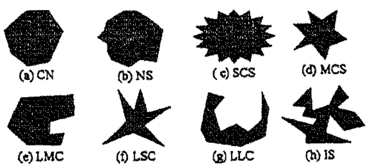


Figure 6. Classification of Polygon Shapes

(6) If the given polygon is LCS shape, firstly SO pattern is used and then the parameter s_7 is checked. When $s_7 > 1$, the polygon should be divided into several CS or NS or SCS shape sub-polygons and s_6 is set. Then the planning method for CS, NS or SCS shape should be applied to each sub-polygon and the compound toolpath pattern and machining procedure are finally obtained.

(7) If the given polygon is LLC shape, firstly SO pattern is used and whether or not the polygon should be divided into CS or NS or SCS shape's sub-polygons is justified. If the given polygon needs to be divided, then it will be separated into two or more regions and the method for planning the toolpath pattern for CS, NS or SCS shape is used. Otherwise. SO, PO and MMO pattern are candidate patterns, in which the pattern with shorter machining time is finally selected as the optimal toolpath pattern.

(8) If the given polygon is IS shape, SO pattern is used and whether or not the polygon should be divided into other seven regular shape types stated above with the bottle-neck-checking method is justified[4]. Then the toolpath pattern planning process for each sub-area respectively by above method is started and the compound toolpath pattern is obtained.

5. EXAMPLE ON GENERATION OF THE OPTIMAL TOOLPATH PATTERN FOR ROUGH MACHINING

In Figure 7(a), a free-form surface with a rectangular boundary generated by UG system is given. For the solid entity defined by such free-form surface. four layers perpendicular to z axis are selected as our research objects, on which the optimal toolpath pattern needs to be created. The shapes of four layers and the created toolpath patterns are displayed in Figure 7(c)~7(f) respectively. In Figure 7(c)~7(f), the areas filled by mesh pattern are non-machining areas, the dash lines are the paths of plunge, retract and rapid traversing movements, and thin lines are the toolpaths created by the reasoning system.

5. CONCLUSION

In the above, a new method for planning the optimal compound toolpath pattern using reasoning system has been introduced. By using such method, high productivity for rough machining could be achieved. This high productivity means considerable production cost reduction, especially for the machining of large and sophisticated parts on expensive CNC machine centre.

Moreover, a method for calculating the area of arbitrary polygon is invented. A parametric classification procedure has also been developed. The simulation result shows many blank cuts could be avoided and machining time could be improved by using such method.

6. REFERENCES

1. H Li, Z Dong and G W Vickers, "Optimal toolpath pattern identification for single island, sculptured part rough machining using fuzzy pattern analysis", Computer-Aided Design, Vol.16, No.11, November 1994, pp787-795
2. Z Dong, H Li, and G W Vickers, "Optimal rough machining of sculptured parts on a CNC milling machine", J. Engng. Ind., ASME, Vol.115, November 1993, pp424-430
3. Yinong Hu, Jun Xia, Youping Chen and Zude Zhou, "An intelligent system for planning machining process of stamping die of car body", Proceedings of the Eleventh International Conference on Applications of Artificial Intelligence in Engineering, R. A. Adey, G. Rzevski and A. K. Sunol, Computational Mechanics Publications, Clearwater, Florida, Sept., 1996
4. Yinong Hu, Theory and practice on the reasoning system for machining the stamping die of car body, Ph.D. thesis, Huazhong University of Science and Technology (1996) (in Chinese)

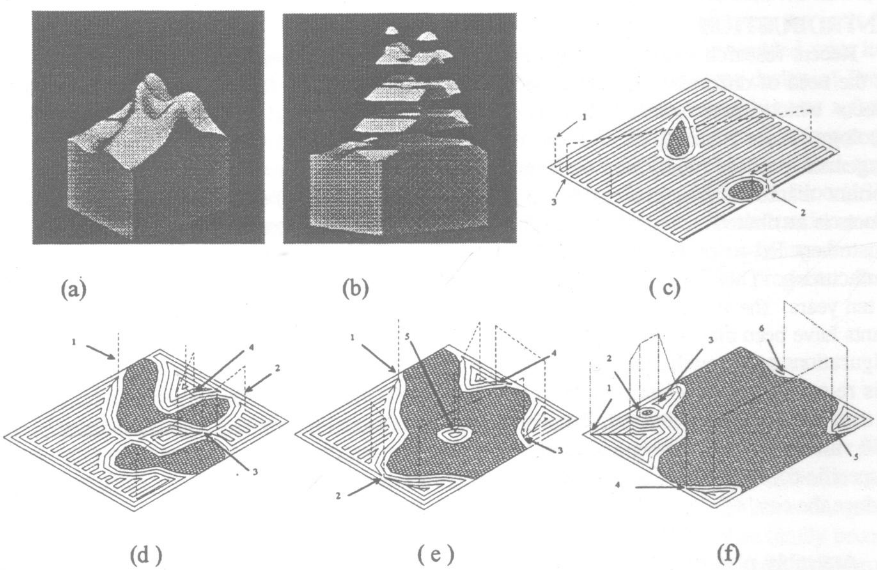


Figure 7. Cutting Layer Shape Analysis and Toolpath Pattern Generation

A Product Family Approach to Assembly Process Planning for Electronics Products

I.S. Fan and C.K. Liu
The CIM Institute
Cranfield University
United Kingdom

ABSTRACT

Assembly is one of the most important manufacturing processes. The requirement for agile response to meet the dynamic demand of customers led to products which are designed for rapid re-configuration and small-batch flexible manufacturing. This is particularly evident in the highly competitive personal computer market. This paper reports on an expert system developed to address electronics assembly process planning. The case example used is a Pentium personal computer. An aggregate assembly plan method is developed to support the assembly line design. Two applications have been implemented in the system : define the product options available for market and to generate the assembly plan for the assembly line for the complete product family.

KEYWORDS

Product Family, Assembly Planning, STEP

1. INTRODUCTION

Recent research work with industry indicates a surge of interest in providing product family to meet the need of different customers. Generally speaking, product family can be defined as a group of products which share something in common, either in physical component or in manufacturing technology. There is strong interest in developing new product description system that can support , management product family and perform aggregate assembly planning. Assembly is one of the most important manufacturing processes. In Europe today, about 40-60% of the cost of manufactured products is attributed to assembly[1]. The requirement for agile response to meet the dynamic demand of customers led to products which are designed for rapid re-configuration and small-batch flexible manufacturing. This is particularly evident in the highly competitive personal computer market. For the past ten years, the average life cycle for PC has been shortened from 4 years to 9 months. The product variants have been increased 4 times as ten years ago. The basic chassis forms the platform for multiple configurations and growth options. Typical production runs are shorter than one week and the assembly line is re-configured on a similarly short time base. SONY dominated the personal audio system market with over 200 models in 1990[2]. A major European personal computer(PC) manufacturer produced up to 150 variants in its assembly line at the same time[3]. Some companies delivered products according to site-specific configurations, a product family approach is necessary rather than a single product in order to reduce the cost[4].

Assembly planning systems for personal computer type electronics products require a different approach from the traditional assembly planning systems. Traditional assembly planning focus on the generation and selection of assembly plans when the assembly structure is established. In the electronics product domain, the basic platform design and overall product assembly structure is known. The detail configuration of the variants are to be generated from the overall structure. There are two roles for the proposed system : to define the product options available for market and to generate the assembly plan for the assembly line. The knowledge rich nature of the assembly planning considerations is best served by the use of knowledge base systems to capture the constraints.

This paper reports on an expert system developed to address electronics assembly process planning. The case example used is a Pentium personal computer to highlight the problem. The product data is stored in the system data base using STEP conformance definition. The bill of material(BOM) for each model, and any variants, are captured as knowledge base. A class of constraints is defined to capture the precedence constraints of each part. Based on the precedence constraints, parts are configured into subassemblies which are built separately before they are integrated in the main assembly. Another set of constraints links the product to the characteristics of the assembly line. This takes into account the re-tooling and setup of the assembly line. The optimal assembly sequence is generated through the automatic evaluation of feasible assembly plans. The commercial development system Nexpert Smart Elements® is used to develop the system interface, rule bases and work as the analysis engine.

2. RELATED WORK

Lin & Chang[5] propose a two-stage planning scheme to generate assembly plans for 3-D mechanical products. A frame-based representation method is used to represent explicitly the non-geometric assembly information. Fan & Liu[6] defined a constraint ratio to measure the influence of the product constraints to assembly planning. Lee[7] proposed a method for the automatic generation of assembly sequences from a liaison graph representation of an assembly through the recursive decomposition of assembly into subassemblies. It automatically identified and avoided those decompositions which incur physically infeasible assembly operations. Delchambre[8] addressed the integration of the assembly process from the initial design of the product to its production. Cho & Cho[9]used a three directional part contact level graph for the automatic generation of assembly precedence constraints for robotic assembly. Assembly precedence constraint was inferred in two steps: the first inferred a precedence constraint for each directional connection by applying the path finding algorithm; the second inferred the precedence constraint for each part to be assembled with its base assembly. SCOPES[10] is the ESPRIT III project 6562: Systematic COncurrent design of Products, Equipment and control Systems. In SCOPES, products had been defined as assembly or as a family of variants. All types and entities for the product were written using EXPRESS formal language part of the ISO standard(ISO 10303, part 11)[11]. Liu and Fisher[12] developed a object-oriented approach using EXPRESS to describe the assembly data and assembly schema. Erens, Mckay & Bloor[13] made use of four levels abstraction to replace the traditional database using three levels abstraction method in product family modelling. Laakko & Mantyla[14] proposed feature-based modelling for product families. Current research dealing with product assembly planning focus on single products. Assembly planning for product family has little been mentioned.

3. GENERATING NEW PRODUCT VARIANTS

It is advantageous to be able to offer new variants of the product family to the market continuously, attracting customer attention and putting pressure on competitors. Products are produced in the product family sharing a common framework but are different to reflect the different taste and preference of customers. The introduction of CAD systems and parametric design eases the generation of variants. A systematic method to manage the generation of variants from the same family becomes an essential tool in both marketing and design. A product data model that represents families of product is developed to support this requirement.

A product family encompasses a great number of similar product models built up from components. In PC, the main components are CPU, mother board, power supply unit(PSU), HDD/FDD, SIMM, casing, and lid(Figure 1).

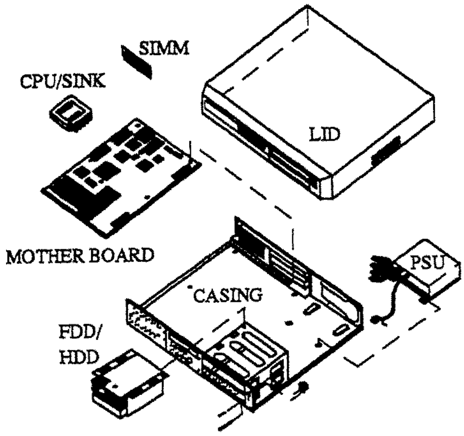


Fig. 1 Exploded view of example product

A class level product definition schema using EXPRESS-G illustrated the relationships among classes(Figure 2). The figure shown in Fig. 2 is part of the product definition adopted in the SCOPES research project which complies with the STEP(ISO standard 10303). Although STEP is still under development, it has widely been accepted that STEP will be the replacement for IGES in product life cycle support and not just for data exchange's specifications. This definition forms the basis of this work.

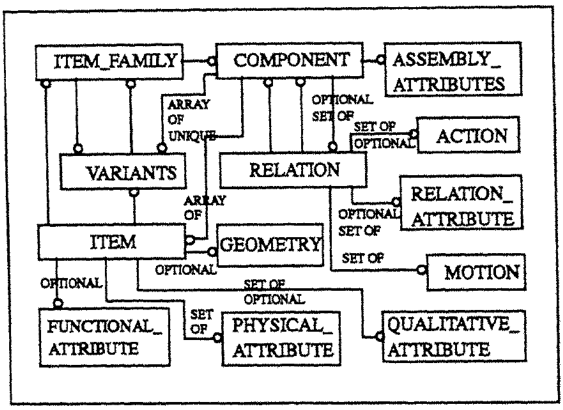


Fig. 2 Product definition schema

Although the actual assembly sequence is not considered in generating new product variants, the related relationship information is required in order to eliminate or reduce downstream assembly planning problems. The following six major items are taken into account when generating product variants.

- commonality: making maximum use of common components within all members of product family
- physical parts: some parts may be specified precisely by the customers.
- special needs: software requirement, warranty scheme, or specify manufacturer for some component.

- industry standard: increased responsibilities of environmental concern may restrict the choice of material in order to comply with the international or local standards.
- abstract concept or ambiguous specifications: when this sort of requirement happen, the system will ask more detail information.
- built-in extension capability: notable example is the extension of memory capacity from standard to user's desire.

The result of generating product variants is a two dimensional table which describes the individual components needed for each variants as in Table 1.

	VARIANT_1	VARIANT_2	VARIANT_3	...
COMPONENT_1(CASING)	STANDARD	STANDARD	STANDARD	...
COMPONENT_2(CPU)	PENTIUM 120	PENTIUM 133	PENTIUM 133	...
COMPONENT_3(MOTHER BOARD)	XXX	XXX-1	XXX-1	...
COMPONENT_4(PSU)	200W	200W	200W	...
COMPONENT_5(CD-ROM)	SIX SPEED	EIGHT SPEED	EIGHT SPEED	...
COMPONENT_6(RAM)	8MB	8MB	16MB	...
COMPONENT_7(HDD)	1GB	1.2GB	1.2GB	...
COMPONENT_8(FDD)	3.5"YYY	3.5"YYY	3.5"YYY	...
COMPONENT_9(LID)	PPP-LID	PPP-LID	PPP-LID	...

Table 1. Product variants and components

The product model supports the handling of special requirements. One example is 8 megabyte(MB) random-access- memory(RAM) in product variant_1 and variant_2 can be extended to 24 MB or more. Since this job can be performed in retailer's shop, it will not be shown in this table. Some customers may require that processors(CPU) are some specified brand to make sure they get exceptional value and quality for the price they pay. This information is stored in each product component's related attributes. In order to comply with ISO 10303, the relationship between parts is represented using EXPRESS formal language. A new product variant will be generated automatically if the customer's requirement is not satisfied. However, any modification for existing product variants is restricted to certain person due to security concern.

4. AGGREGATE ASSEMBLY PLAN

With the definition of relationship among components, the precedence constraints can be generated and assembly process planning can proceed. Traditional method dealing with assembly planning focus on automatically generating assembly sequence for single product. At present, current solutions to assembly line design handling with product family is through the use of modular design or form factor method to standardise the contact interface among components. However, the design flexible is constrained due to the adoption of modular design or form factor method. In single product assembly planning, the main consideration are attributed to product's precedence constraints(geometric and non-geometric constraints)[15]. In product family assembly planning, component variants is also one of the major concern due to the interaction affect. An aggregate assembly plan(AAP) method is developed to support the assembly planning in the case of product family. The idea behind the AAP is to generate generic assembly plan through the product family table for assembly line design. As in Table 1, each product variant consists of same or similar components. The changes at component level would or would not influence assembly planning depend on whether it belongs to an 'assemble affected' item or 'non-assembly affected' item. For example, colour is a factor causing product variant However, it would not affect the assembly plan and is classified into 'non-assembly affected' group. Those parameters affecting the assembly plan are called 'assembly affected' attributes. A product family table is built and used to link and analyse 'assembly affected' attributes for product family assembly process planning. The inference rules for above analyse function are established in the system's knowledge base. The algorithms for AAP are listed as followed:

- Define the parent product(standard product).
- A product family table is built through the product variant table.
- Analysed and summary the interaction between product variants and indicates all contributors, geometric and non-geometric.
- Identify assembly-affected parameters.
- Link all the assembly-affected parameters within the product family.
- Translate assembly-affected attributes into the precedence constraints.
- Estimate the assembly complexity by Constraint Ratio Method[6].
- Perform the assembly process planning for the product family.
- List and suggest the optimal aggregate assembly plan.

This aggregate assembly plan is the basis for the design and layout of the assembly line. Every time a product variant is generated or changed, the system will check all its components attributes. The assembly-affected parameters are entered into the database and linked to parameters within other product variants. This enables the assembly planner to account for complex interaction when assembly planning is proceeding.

5. CONCLUSION

It is commonly accepted that companies offer families of product is a critical factor to compete in the fierce marketplace. In this paper, a product family approach to assembly process planning for electronics products has been presented. Currently, we have built the system architecture and defined the product component and their relationships using the EXPRESS formal language in Nexpert Smart Elements® tool. The system can generate the product variants to match different customers' requirements. It brings together a comprehensive range of product variants listed in a 2-D table. Furthermore, AAP method is developed to support the assembly process planning in the case of product family. The product family table as well as assembly-affected attributes of each product variants are used to provide the useful information in the assembly planning process. The algorithm for AAP has been given in list. Future work will be concentrated on the dynamic rebalance of worker in an assembly line for PC products and enhance the user interface by using Open Editor® tool.

6. REFERENCES

1. Wallace, G., "An integrated approach for the application of design for assembly" , Phd Thesis, Cranfield University , U.K., 1995.
2. Wheelwright, S.C. & Clark, K. B., "Creating project plans to focus", Harvard Business Review, Mar-Apr, pp.70-82, 1992.
3. Fan, I.S. & Liu, C.K., Private interview with a major European PC manufacturer, England, 1995.
4. Van der Linden, F.J. & Muller, J.K., "Creating architectures with building blocks", IEEE Software, **12**(6), pp.51-60, 1995.
6. Fan, I.S. & Liu, C.K., "Constraint Ratio in Product Assembly Planning", International Journal of Advanced Manufacturing Technology, (Paper accepted for publication.)
5. Lin, A.C. & Chang, T.C., "An integrated approach to automated assembly planning for three-dimensional mechanical products" , International Journal of Production Research, **31**(5), pp.1201-1227, 1993.

7. Lee, S., "Subassembly identification and evaluation for assembly planning", IEEE Transactions on System, Man, and Cybernetics, **24**(3), pp.493-503, 1994.
8. Delchambre, A., Computer-aided assembly planning, Chapman & Hall, London, 1992.
9. Cho, D.Y & Cho, H.S. "Inference on robotic assembly precedence constraints using a part contact level graph", Robotica, **11**(2), pp.173-183, 1993.
10. SCOPES Project 1992-95, ESPRIT III 6562 Systematic Concurrent Design of Products, Equipment and Control systems.
11. ISO 10303 Industrial automation systems and integration--Product data representation and exchange, Part 11: Description methods: The EXPRESS language reference manual, 1994.
12. Liu, T.H. & Fischer, G. W.," Developing featured-based manufacturing applications using PDES/STEP", Concurrent Engineering , **1**(1), pp.39-50, 1993.
13. Erens, F.J. & McKay, A. & Bloor, S., " Product modelling using multiple levels of abstraction instances as types", Computer in Industry, **24**(1), pp.17-28, 1994.
14. Laakko, T. & Mantyla, M., "Feature-based modelling of product families", ASME International Computers in Engineering Conference, pp.45-54, 1994.
15. Liu, C.K. & Fan, I.S., "Integrated product definition Model for Electronics Board Level Assembly Process Planning", Proceedings of the 12th International Conference on CAD/CAM Robotics and Factories of the Future, pp.884-889, London, 1996.

NEURAL NETWORK CAPP SYSTEM MODELING

Chen Bingkui
The State Key Lab of Mechanic Transmission,Chongqing University, Chongqing 630044 China
Wang Dehua
Guizhou Aviation Group Yuntian Cooperation, Guiyang 550206 China
Lin Tengjiao
The State Key Lab of Mechanic Transmission,Chongqing University, Chongqing 630044 China

ABSTRACT

The thinking mode involving in the computer aided process planning is studied. Artificial neural network technology is reviewed. A mechanic CAPP system model based on neural network is putforward. Features of the system and possible developing methods are discussed.

KEYWORDS

Neural network Computer aided process planning Artificial intelligence

1. INTRODUCTION

Process planning is an important link between design and manufacture, which relates to various factors such as machine tools, management and even workers Computer aided process planning technology becomes a forceful tool in mechanic industry which is the fundamental and key issue in developing computer integrated manufacturing system. Most practical Computer Aided Process Planning(CAPP) systems are developed by variant approach. Although many scholars are devoted to generative CAPP technology, few successful systems are reported It has been drawing more and more attention both at home and abroad to develop an intelligent CAPP system that can accumulate expertise and generate planning codes.

In this paper, we will analyze the thinking behaviors during the process planning, which is useful to lay bare the essence of process planning. Artificial Neural Network(ANN) technology will be reviewed and a mechanic CAPP system model based on which will be presented.

2. ANALYSIS OF PROCESS PLANNING

For an experienced engineer, manufacturing processes of many machine parts are "experienced", which means the data are stored in this brain. Although details in every working procedure might not be so clear, process course of a part is ready in his mind. If he undertakes a planning task, he will retrieve the existing data in his memory at first. If there are similar parts, he will find out the information somewhere and revise the planning until it satisfies the concrete requirements. Third, complete the whole planning. If there are not similar ones, he will create a new process.

In view of philosophy, the process planning involves two inter-acting thinking modes. One is logic thinking and another is image thinking. In the first step, existing data in an engineer's brain being an image of the past planning(experience), the engineer can make his choice immediately. So image thinking is in the lead which is rapid and figurative. In the second step, the planer will pick out the most similar one and determine parameters according to the design drawing. He will work out procedures, choose machine tools, design fixtures and so on. It is apparently that the engineer mainly "thinks in logic mode" at this step. After that, the planer finishes his concrete planning including procedures and technical documents.

As we know that some advanced CAPP systems can generate manufacturing process code, but these systems are limited to some specific and simple applications, say, shafts. In addition, they can not accumulate expertise for future purpose. Since these systems are mainly developed by logic inference method, experts are puzzled by difficult problems such as combination explosion with the increase of inference rules and solution scale. Another point to be notified is that design information is the foundation of process planning. So, the unique representation of design drawing is the bridge of design and process planning.

It is well known that a practical intelligent CAPP system should resemble human brain that can "think" both in logic and image. Traditional digital computer can simulate logic thinking wonderfully, but it behaves poorly in image thinking mode. Therefore, new technology and method should be introduced to break through the straits.

3. ARTIFICIAL NEURAL NETWORK

Artificial Neural Network(ANN) model inspired by the biological nervous system follows the same basic design: neurons connected to many other neurons forming a neural network. These systems can intrigue brain functions such as classification, associative memory, learning and accumulation of expertise. They are robust, and can simulate image thinking of human brain. As in computer science, advancements in ANN are progressing at a rapid pace. In current conventional ANN applications, jobs of loan officers and stock market experts are attempted. Industrial applications are growing also in the areas as structural optimization, diagnostic system and expert system.

There are about forty ANN models developed in the light of specific applications. They can be classified into supervised and unsupervised learning models. Adaptive Resonance Theory(ART) is a typical one of these, which adopts competitive learning self-steady mechanism. It is characterized by: (1) it can learn on line for any input vectors; (2) learning process is self-organized and unsupervised; (3) learning efficiency is relatively higher that feed-back NN model.; (4) it can avoid local minimum problems. ART model has two layers: input and output layer. The topology of ART1 neural network is diagrammed in figure1.

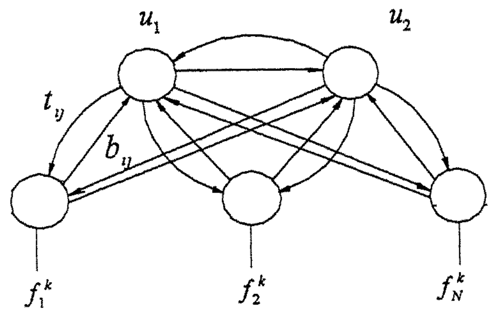


Fig.1 Topology of ART1

—The learning algorithsm of ART1is as following:

(1) Iniatiallizing weighting matrix. If the network has learned, collect the connection weights and turn to step2 Connection weights from top to bottom:

$t_{ij}(0) = 1$

Connection weights from bottom to top:

$b_{ij}(0) = 1/(N+1) \quad 1 \le i \le N, 1 \le j \le M$

where N is the dimension of sample vector; M is the maxium classification family.

(2) Input the sample number k and its feature vector as well as ρ:

$f^k = (f_1^k, f_2^k, \ldots, f_N^k)$, $f_i^k \in \{1,0\}$

and

$0 \le \rho \le 1$

(3) Calculation of matching state:

$u_j = \sum_{i=1}^{N} b_{ij}(k) f_i^k$, u_j is the output of net j. u_j=*Max* is the best matching state.

(4) Testing of alerting threshold:

$|F^k| = \sum_{i=1}^{N} f_i^k$, $|t_{ij} * F^k| = \sum_{i=1}^{N} t_{ij}(k) * f_i^k$

if $\frac{|t_{ij} * F^k|}{|F^k|} \geq \rho$, turn to step6

(5) Set $S(u_j = max) = 0$, Re-matching until threshold is greater or equeal to ρ;

(6)Adjusting weights of best matching family nets:

$t_{ij}(k+1) = t_{ij}(k) * f_i^k$,

$$b_{ij}(k+1) = \frac{t_{ij}(k) * f_i^k}{1/2 + \sum_{i=1}^{N} t_{ij}(k) * f_i^k}$$

(7)Next sample.

Realization of the algorithms presented above is not difficult for most experienced programmers. The outstanding properties of ART1 neural network is suitable to classification cases especially.

4. REPRESENTATION OF A PART AND ITS PROCESS

Structure, matieral and other design parameters are decisive fators in process planning. For a factory, technical process is almost dependent on design drawing, though some procedures might be determined by planers' preference. If we establish the representation method of a part, we can develop a highly non-linear map relationship between design and process $f.R^n \to R^m$ In this way, retrieve of process planning can be realized by classification of a part.

There are many methods to classify machine parts. The most commonly used one is Group Technology (GT). As we know, GT acts according to the similarity of machine parts. Although GT data is unsuitable for neural network, the code number being 0~9, theory of which will be a foundation of developing a method to represent a machine part and its process.

Information of a machine part includes many aspects such as design parameters, structural features and manufacturing processes, which are called "knowledge" in terms of Artificial Intelligence(AI). Full representation of the knowledge with pure AI method is rather difficult. Since the dominant data of a machine part is limit, ingenious organization of which would bring about satisfactory results. Here, a vector is proposed to describe a mechanical part.

The vector has 100 or more components depending on the scale of CAPP system. More components mean more precise representation. The values of components are set to 0 or 1. These components are divided into three sections: the first section representing part family such as shafts, cases, plates, etc.; the second section representing concrete form features such as material, precision, fillet, dimension, quench and so on; other components are third section representing factors associate with manufacturing process. Number of components in every section will be determined according to specific situations. So every machine part can be expressed as a unique vector.

For example, as a gear is concerned, the family of the gear belongs to is obviously and the first section is given out easily. The module, tooth number, precision, material structural features and other design factors form the second section. Manufacturing process factors relates with equipments, process planning and workers, which is supplement information. Through inter-dialog process, the vector of this gear is determined.

According to the specific component value of the first section, a part's family is logically defined. This is a rough classification of a machine part. Usually, this job is done by a sub-expert system. The rest components of the vector are put into ART1 neural network. The ART1 will classify the data and store the result.

5. NEURAL NETWORK CAPP SYSTEM

According to the discussion above, we construct a CAPP system based on neural network. The system model is shown in figure 2.

When we begin to plan a new process, the interface of the CAPP system will prompt us to input initial destination data. Then, the system will process the data and form a new expect vector. The vector will be put into ART1 neural network, which will retrieve the most similar exist design vector. In the mean time, the system will seek out corresponding working process. This procedure is similar to image thinking of an engineer. Usually some modifications are needed to satisfy the design specifications. Next step, the CAPP system will complete the process planning.

On the other hand, when the new process planning is finished and saved, computer will prompt us to do some further dialogs. This process will record the new characteristics of the planning and transform the data into neural network. ART1 will learn, classify and store the new "knowledge", that is, accumulating expertise.

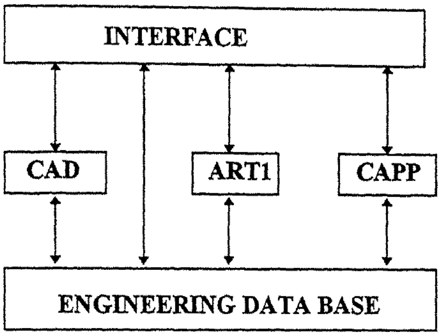


Fig.2 Neural network design system model

A shortcut to develop such intelligence system is to reform the CAPP system being used. This method inserts the ART1 neural network module to the existing system , and programs common joints among other modules. We can also purchase a neural network hardware and construct a new CAPP system.

It is notified that in this preliminary neural network CAPP system, exchange of information among neural network, CAD and CAPP sub-system is realized by person-computer dialogs, which is an indirect connection and engineer should interfere with the program. In a highly intelligence system, the connection will be direct, which means complete drawing, complete learning, complete process planning. Such a system is our efforts will be devoted to.

6. CONCLUSION

Process planning involves logic and image thinking mode. Digital computer and neural network can simulate the two thinking modes respectively. A preliminary neural network process planning system model is presented and some interest issues are discussed. The contributions in this paper provide a new idea to develop a practical intelligent CAPP system.

7. REFERENCES

1 L.Berke, S.N.Patnak, P L.N.murthy. Optimum Design Of Aerospace Structural Components Using Neural Network. *Computers & Structures*, 1995,48(5):1001 ~ 1010

2 H.Adeli, Hyo Seon Park. A Neural Dynamics Model For Structural Optimization-Theory. *Computers & Structures*, 1995, 57(3):383 ~ 390

3 Hyo Seon Park, H.Adeli. A Neural Dynamics Model For Structural Optimization-Application to plastic design of structures. *Computers & Structures*, 1995, 57(3):391 ~ 399

An Optimal Expected-Time Algorithm for Voronoi Diagram of Disjoint Polygonal Objects

Wang Jia ye
Gintic institute of Manufacturing Technology
Nanyang Technological University,
Singapore
Department of Computer Science
Shandong University
P.R.China

Wang Wenping
Department of Computer Science
The University of Hong Kong
Hong Kong

Liu Ding Yuin
Department of Mathematics
Fudan University
P.R.China

ABSTRACT

The Voronoi diagram of a set of given objects is a geometric structure partitioning the space of the objects into disjoint regions. The structure assigns every point in the space to its nearest object. From the structure one can retrieve the points in close proximity to the objects. A variety of problems in computational geometry can be solved based on the Voronoi diagram. These problems include some very important applications in industries. For example the cutter path in NC milling, the largest empty circle in PCB mold design etc. can be computed immediately from the Voronoi diagram.

The Voronoi diagram of a set of points in the plane was studied much earlier than Voronoi diagram of polygons. The former was first introduced by Voronoi in 1908, and has been studied by researchers in several fields. Shamos and Hoey presented an $O(n\log n)$ algorithm in [1]. An $O(n)$ optimal expected time algorithm was published in [2]. Both of these papers study the Voronoi diagram of a set of points. Lee [3] presented an $O(n\log n)$ divide-conquer algorithm for computing the Voronoi diagram of a set of simple polygons. Yap came up with a sophisticated worst-case optimal algorithm [4].

In this paper the objects discussed are polygons with sides being line segments. The Voronoi diagram of a set of polygons in the two dimensional Euclidean plane is studied. A new algorithm is presented that computes the Voronoi diagram in $O(n)$ expected time. The algorithm can be extended to the case where the sides of the polygons may be curve segments. Since the algorithm is based on constructing Voronoi polygons of line segments and vertices individually, it is simple and easy to implement.

KEYWORDS

Voronoi diagram, Algorithm, Complexity.

ALGORITHM

The plane is divided by each polygon into an infinite region and a finite region, either of which can be chosen as the outer region of the polygon. The Voronoi diagram is only created in the intersection of the outer regions of all the polygons. The geometric points and their construction mentioned in the following discussion are restricted to be in the intersection region. When we refer to the "distance" between two points measured along their connecting line segment, this line segment must also be in the intersection region. In the following description, the distance from a point to a line segment means perpendicular distance, and the intersection point between the perpendicular line and the line segment must be on the line segment.

Two definitions are needed for the new algorithm:

Definition 1: The **Voronoi polygon** (VP) associated with a side is the locus of the point such that its distance to the side is not greater than its distance to any other sides or vertices.

We can also define the Voronoi polygon associated with a convex vertex of a polygon.

Definition 2: The **Voronoi polygon** (VP)' associated with a vertex is the locus of the point such that its distance to the vertex is not greater than its distances to any other sides or vertices.

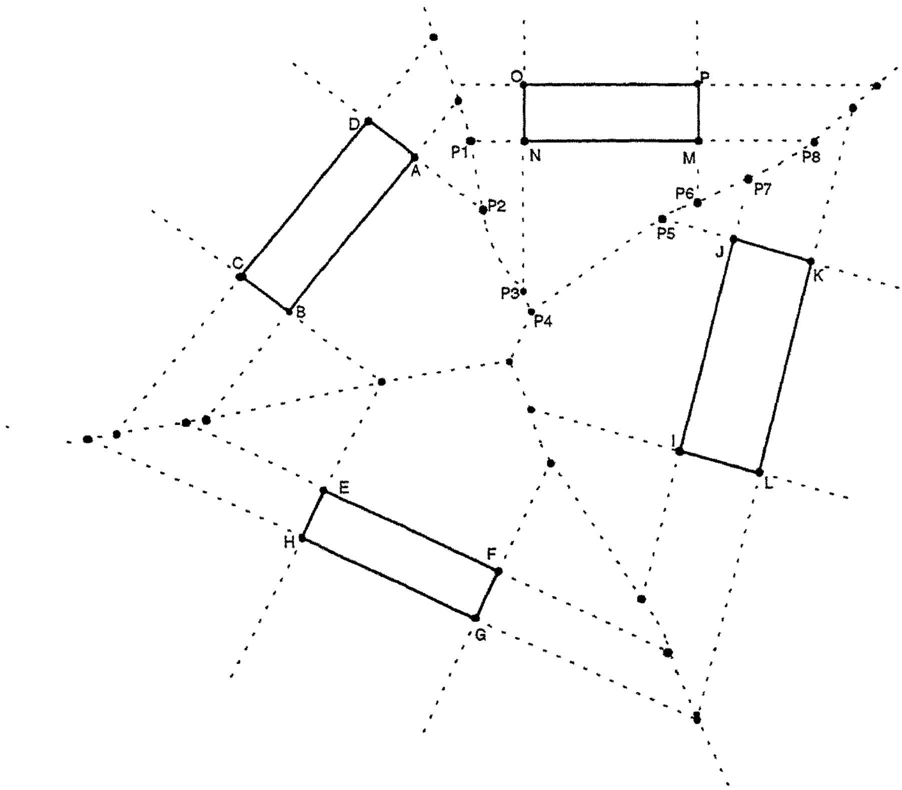


Figure 1

Voronoi polygons partition the plane into a collection of disjoint regions which we shall refer to as **Voronoi diagram** (ref. Figure 1). The sides of the **VP's** are called the sides of the Voronoi diagram. Each side of the Voronoi diagram, except those on the objects, is common to two **VP's**. Any point on a common side of the Voronoi diagram has the same distances to the sides or vertices that share the common side. Voronoi sides can be generated in the following three cases.

1. A bisector of two sides (e.g., in Fig. 1, P4P5 is the bisector of sides MN and IJ).
2. The bisector of two vertices (e.g., in Fig. 1, P1P2 is the bisector of vertices A and N).

3. The trace of points with same distances to a vertex and a side (e.g., in Fig. 1, the distance of any point on P2P3 to vertex N is same as the perpendicular distance of side AB to that point).

The trace in case 3 is a parabola. A **VP** associated with a side is restricted within two rays perpendicular to the side from its two end points (e.g., NP3 and MP6 for **VP** associated with side MN in Fig. 1). A **VP** associated with a vertex is restricted within the two rays perpendicular to the adjacent sides of the vertex (NP1 and NP3 for **VP** associated with vertex N in Fig. 1). The region restricted within the two rays is called the *restricted region* associated with a side or a vertex (**RR**). The two rays are called rays attached to the side or vertex.

The above three cases can generate an infinitely extended line or parabola. But only one or several segments of the line or parabola may become the sides of a **VP**, and these segments must be in two corresponding **RR**'s. For example, the trace of points with same distances to vertex N and line AB is a parabola. But only P2P3 being in both the **RR**'s associated with vertex N and line segment AB becomes the side of **VP**'s associated with N and AB. Similarly P4P5 is part of the bisector of line MN and IJ in both the **RR**'s associated with MN and IJ. From now on we will refer the part of line and parabolic being in two corresponding **RR**'s as a *bisector.*

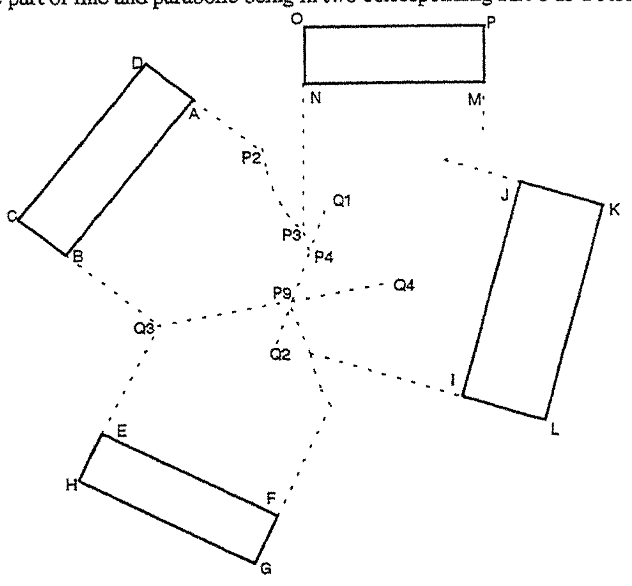


Figure 2

The following lemmas can be proved.

Lemma 1. The **VP** is a closed polygon or an open polygon with only two infinite sides.

Lemma 2. The **VP** associated with side S is the smallest polygon whose sides consist of side S, the two rays attached to S, and bisectors between S and other sides and vertices.

Lemma 3. The **VP** associated with a vertex P is the smallest polygon whose sides consist of the two rays attached to P and bisectors between P and any other sides and vertices.

The algorithm presented is based on Lemma 2 and Lemma 3. For a side (or vertex) we create its associated **RR**, and the bisectors between the side (or vertex) and all other sides and vertices. To create the smallest polygon we compute the intersection points of the bisectors and the intersection points between bisectors and the two rays attached. If there is an intersection between two bisectors, each bisector may be divided into two line segments. The segments far from the side (or vertex) are eliminated. For example, when the **VP** associated AB is created, bisectors Q1Q2 and Q3Q4 intersect, and the segments P9Q4 and P9Q2 being far from AB are eliminated. If the retained segments intersect with another bisector, parts of them may also be eliminated; Q1P4 in Fig. 2 is an example. Eventually, by Lemma 1, we can find a smallest closed polygon or a polygon with two infinite sides. The bisectors and rays that are not the sides of the smallest polygon will not affect the **VP**.

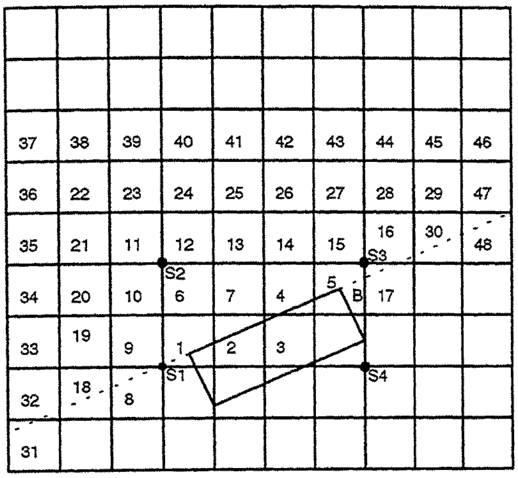


Figure 3

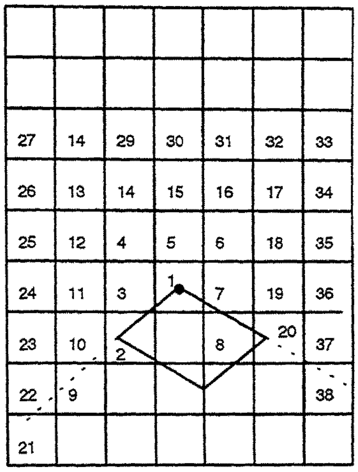


Figure 4

To create a **VP** associated with a side (or vertex) the bisector of the side (or vertex) and all other sides and vertices, as well as their intersection points, must be computed. However, computing the intersection of line segments is time consuming with the simple pair-wise testing approach. Other theoretically worst-case optimal algorithms are complicated and difficult to implement; furthermore, they may not be efficient for a problem of reasonable size in practice, or a problem with the input objects being nearly uniformly distributed, as will be considered in our work. In the following, exploiting the uniform distribution and locality of input polygonal objects, we present an optimal expected-time $O(n\log n)$ algorithm based on local search.

Obviously, a **VP** associated with a side (or vertex) is determined by the local objects near the side (or vertex). Therefore creating a **VP** can be carried out by searching the objects surrounding the side or vertex in an expanding fashion. To facilitate the search, the plane is first partitioned by a regular grid into small squares (cells). Figure 3 and Figure 4 show how the spiral like search from cell 1 might proceed for creating **VP**'s associated with a side or a vertex, respectively. Numbers labeled in the cells denote the order in which the cells were visited.

To create a **VP** associated with a side, the search starts from the smallest rectangle containing the side (ref. S1S2S3S4 in Fig. 3), then proceeds layer by layer a spiraling manner. When a new object is met, we compute the bisectors of the side and the new object's sides and vertices, and compute the intersection points of these bisectors with the retained segments of the previous bisectors. After several objects have been searched and processed, a closed or open polygon with two infinite sides (called potential **VP**) is created. We can then find a roughly double sized polygon containing the potential **VP** such that any objects outside the double-sized polygon will not affect the construction of the potential **VP**. If the cells traversed by the spiral like search are all outside the double-sized polygon, the search can be terminated.

The analysis of the complexity of the algorithm is similar to the discussion in [2]. In the following discussion, we assume that the objects are in a unit square, and Voronoi polygons are closed. Note that the **VP** can be open only for sides or vertices near the boundary of the unit square; so we stop the search when reaching the boundary of the unit square, since only the **VP**'s in the square are generated. The complexity of creating the **VP** can still be derived with Lemma 4 and Lemma 5.

If one wants to create the whole Voronoi diagram, the same strategy as in [2] can be applied. That is, the vertices and sides are divided into inner and outer sets. For the outer vertices and sides an O($n$log$n$) worst-case algorithm is applied. Since the expected number of outer vertices and sides is $O(\sqrt{n})$, the expected time of creating is Voronoi diagram is O(n).

Lemma 4: If the n vertices of the polygons Pi $(0 \le i \le m)$ are chosen independently from a uniform distribution on a unit square and the maximum length of the edges of the polygons is less than $C_0/\sqrt{n}$, where C_0 is a constant, a Voronoi polygon associated with each side can be generated in O(1) expected time.

Proof.

Assume that the area of the small square (cell) is C/n. There are n/C cells, so that the expected number of points in each cell is C. Let

$q=C/n$.

The probability that a certain fixed cell is not assigned a particular point is $(1-q)$, and the probability that a particular cell is empty is

$$(1-q)^n=(1-C/n)^n<e^{-C} \qquad (1)$$

So the probability that k given cells are all empty is bounded above by e^{-Ck}.

Suppose AB is the side whose associated **VP** is being creating. Assume that the smallest rectangle containing AB is S1S2S3S4 (ref. Figure 3). Let S1S2 and S2S3 span M_1 and M_2 cells respectively. The number of cells on the first layer is no greater than M_1+M_2+4. The number of cells increased for extending each layer is not greater than 4. After k layers have been searched, the number of total cells searched is not greater than $k_1=(M_1+k)(M_2+k)$, where $k<\sqrt{n/C}$.

If the polygon created is still open after k layers of cells have been searched, it means that we can find a point P in the restricted region associated with AB, and there is no bisector intersecting the perpendicular line form point P to AB, and the distance from P to AB is not less then $k\sqrt{C/n}$, where $\sqrt{C/n}$ is the length of the side of cell. By the appendix, there are no vertices of any objects in the intersection region of the searching area and the circle centered at P with radius $k\sqrt{C/n}$. The area of the intersection region is not less than the area of quarter of the circle. That is, after k layers of searching, more than

$$k_2=[0.25*\pi*((k-2)*\sqrt{C/n})^2/(C/n)]=[0.25*\pi*(k-2)^2] \qquad (2)$$

cells are found to be empty. In above expression [x] is the largest integer being less than x. From (1) the probability of such as event is less than e^{-C*k2}. Because the n points are chosen independently, the expected number of the vertices being placed into the non-empty searched cells is not greater than

$$k_3=\frac{n(k_1-k_2)}{n/C-k}=\frac{C(k_1-k_2)}{1-k_2(C/n)},\quad n/C-k_2>0 \qquad (3)$$

So the complexity of searching the k_1 cells and to form the potential **VP** for the side AB is less than $C_1 * (k_1 + k_3^3)$, where C_1 is a constant being independent of n. The total expected cost to create the **VP** for side AB is

$$T(n) = \sum_{0<k<\sqrt{n/C}} C_1 * (k_1 + k_3^3) * e^{-C*k2} \quad (4)$$

It can be proved that $T(n)$=O(1) by the convergence of the series $\sum_i i^m * e^{-i}$ for any integer m. This completes the proof.

Lemma 5: If the n vertices of the polygons P_i ($0 \le i \le m$) are chosen independently from a uniform distribution on a unit square and the maximum length of the edges of the polygons is less than $C_0 / \sqrt{n}$, where C_0 is a constant, a Voronoi polygon associated with vertex can be generated in O(1) expected time.

The proof of the lemma is similar to Lemma 4. By Lemma 4 and Lemma 5, we obtain the following Theorem.

Theorem: If the n vertices of the polygons P_i ($0 \le i \le m$) are chosen independently from a uniform distribution on a unit square and the maximum length of the edges of the polygons is less than $C_0 / \sqrt{n}$, where C_0 is a constant, the Voronoi diagram can be generated in O(n) expected time.

APPENDIX If the trace of points with same distances to Q and AB does not intersect the perpendicular line from point P to AB, point Q is outside the circle centered at P being tangent to line AB.

Proof.

Assume that the circle is tangent to AB at C. EF, the bisector of points Q and C, intersects line PC at D. The trace of the point with same distances to Q and AB must pass point D. If the intersection point of the trace and PC is not between P and C, D is not between P and C too. Obviously, D is not located between P and C if and only if Q is not in the circle.

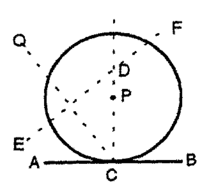


Figure 5

REFERENCES

[1] M.I.Shamos and D.Hoey. Closest-Point Problems. In *Proc. 16th Annual IEEE Symposium on Foundations of Computer Science*, October 1975.

[2] Jon Louis Bentley, Brouce W. Weide and Andrew C. Yao. Optimal Expected-Time Algorithms for Closest Point Problems. *ACM Transactions on Mathematical Software* 6(4): 563-580,1980.

[3] D.T.Lee. Medial Axis Transformation of a Planar Shape. *IEEE Trans. Pattern Analysis and Machine Intelligence, PAMI* 4(4):363-369, 1982.

[4] Yap. An O($n$log$n$) Algorithm for the Voronoi Diagram of a Set of Simple Curve Segments, *Discrete and Computational Geometry* 2(4): 365-393, 1987.

[5] Martin Held. On the Computational Geometry of Pocket Machining, Springer-Verlag, 1991.

THE BENEFITS OF COHERENT EXPLOITATION IN CSG RENDERING ALGORITHMS

W Kurdthongmee
Department of Computer Science and Information Systems,
Brunel University
United Kingdom.

ABSTRACT

Most techniques that improve the speed of CSG image generation rely heavily on the exploitation of some form of coherence in the object space, image space or model structure of the primitive solids. However, any overview of the various forms of coherence and their exploitation, along with the relationships between them, and some indication of tangible benefits, is missing. In this paper, we give the overview of coherence and propose the experimental technique for evaluating the benefits of exploitation of coherent properties in CSG rendering algorithms. Our technique is based on evaluating the rendering time of algorithms with simulated CSG models which have variations in number of primitives, type of primitive, and complexity of primitive, with and without the existence of coherence.

KEYWORDS

Coherence Exploitation, Rendering Algorithms, Solid Modelling, CSG.

1. INTRODUCTION

Although, there are a number of schemes to represent three-dimensional solid objects unambiguously, two representation schemes are by far the most widely used. They are:

-the constructive solid geometry, CSG, that describes a volume as a number of primitive solids related by a set theoretic expression,

-the boundary representation (B-Rep) that represents the boundary of an object in a topologically consistent form.

Many papers have been written about the relative merits of the two representations [11, 13, 16, 21]. The boundary representation has in the past been more widely used, but the constructive solid geometry offers a number of advantages, such as numerical stability and the simplicity of algorithms, that make it worth pursuing. A major disadvantage of CSG representation is that it is a global definition of a solid: that is to say, the complete composite solid has to be evaluated to determine whether any point is inside or outside of the composite solid. There is no exception for this disadvantage for CSG rendering algorithms.

Fortunately, the global definition of CSG models can be reduced by exploiting some form of coherence [2, 3, 4, 17, 21].

In Computer Graphics, the foundation of nearly all efficient algorithms for image generation is the utilisation of a coherence property. The clearest definition of coherence is: "*Coherence is a degree to which parts of an environment or its picture or projection exhibit local similarities.*" This definition follows from the fact that the properties of objects such as depth, colour, and texture vary smoothly from one point to a neighbouring point. The exploitations of coherence properties in image generation involves reuse of the information that has already been computed for one part of an image to the nearby part without recalculating from scratch.

There are many different types of coherence properties utilised in rendering algorithms and these are outlined as follows:

1. Object-space coherence an object tends to be continuous and different objects tend to be disjoint in three dimensional object space., i.e. if a point is occupied by some object, the neighbourhood of the point is probably also be occupied by the same object.

2. Image-space coherence an object tends to be continuous and different objects tend to be disjoint on a two dimensional screen after projection. Image space coherence includes scanline coherence, which means consecutive scanlines are probably nearly the same, and pixel coherence, which means consecutive probably have nearly the same colour.

3. Frame-to-frame (temporal or modification) coherence an image plane tends to be the same and from frame to frame there is only a little change in a set of objects displayed.

4. Algorithmic dependent coherences

4.1. Ray coherence rays which have the same origin and nearly the same direction will probably intersect the same object at nearly the same location.

4.2. CSG coherence for adjacent rays, the order in which primitives are encountered along the ray will often be the same, and hence the classification of intersection points will always be the same. Therefore, repeated classification of the same sequence of intersection points can be avoided by storing the ray classification for each sequence in a suitable data structure [11].

Well-known direct rendering algorithms for CSG models are the ray casting algorithm, the scanline hidden surface algorithm and the depth buffer [1, 16, 18]. Although the ray casting algorithm gives the most attractive image quality, it is computationally expensive. The depth buffer algorithm is only feasible using special-purpose hardware. Atherton's scanline hidden surface algorithm gives reasonable rendering times on a general-purpose workstation, however, its drawback is the use of polygonal primitives. The improvement of these algorithms has been done by utilising some forms of coherence to increase their efficiency (i.e. [2, 5, 15, 17, 20]).

This paper gives the overviews and benefits of the exploitation of coherent properties in rendering algorithms and proposes the method for evaluating their relative performance. The paper is organised as follows. We start by giving an overview and benefit of coherence exploitation in well-known rendering algorithms. Then, we present an experiment for evaluating the benefits of coherence exploitation.

2. AN OVERVIEW OF COHERENCE EXPLOITATION

Much work has been done on improving the performance of rendering algorithms by exploiting coherence properties. In this section, we give the overview of such exploitations.

2.1 Exploitation of an Object Space Coherence

Most object space coherence has been exploited in the ray tracing algorithm. Early techniques focused on utilising bounding sphere to reduce the calculation of ray/object intersections. Unfortunately, the improvement is only the order of a constant. Other approaches [6, 7] use the partitioning of the object space into several subspaces. Using object space coherence, an object is expected to belong only to a number of subspaces. Thus the actual ray/object intersection checks are needed only for the objects which are contained in subspaces which intersect with the ray. The efficiencies of the algorithm depend on how well the object space is divided. [6] divides the entire space uniformly into congruent cubes, which results in a better performance than the octree method [7]. Ohta [14] suggested that, by increasing the number of subspaces, the number of the objects which need to be checked is reduced. But at the same time, the number of the subspaces which intersect with the ray increases. Therefore, it is concluded that the improvement is somewhat limited.

2.2 Exploitation of an Image Space Coherence

Image space coherence also includes scanline coherence, pixel coherence, depth-coherence and span boundary coherence. All of these have been exploited in nearly all rendering algorithms in the scanline-oriented category [1, 5, 9, 13, 15]. [20] introduces a concept of picture-plane coherence, that is a form of image space coherence, and is used in reducing set theoretic expression in rendering and calculating integral properties of CSG models. In the Z buffer rendering class, [19] propose the ZZ buffer algorithm for rendering CSG models. Following this technique, image space coherence is exploited to reduce the CSG tree to be localised to the set of primitives in the local image area. Recently, [8] proposes algorithm called "*hierarchical z-buffer visibility*" for rendering a geographic database. The algorithm uses two hierarchical data structures, an object space octree and hierarchical of z-buffer on an image space, to accelerate scan-conversion. The author claims that the two hierarchical data structures make it possible to reject hidden geometry rapidly while rendering visible geometry with the speed of scan-conversion.

2.3 Exploitation of a Frame-To-Frame Coherence

The frame-to-frame coherence has been proposed to speed-up the rendering process in animation applications [5, 8].

2.4 Exploitation of an Algorithmic Dependent Coherence

In addition to general coherence properties, there are two coherence properties which are algorithm-dependent. The first, introduced specifically for CSG representations is called CSG coherence [11] and was first used in ray casting algorithms. It follows from the fact that the same sequences of intersected object primitives together with their status (in, on, or out) with respect to the intersected primitive always give the same composite solid classification. This property can be used in reducing the number of CSG classifications of the same ray/primitive intersection sequences in CSG point classification algorithm. That is to say the problem is reduced from re-classifying all ray/composite-solid sequences of all rays to comparing the sequence of the current one with the previous classification stored in a suitable data structure. The second algorithmic-dependent coherence is ray coherence. By definition, ray coherence usually causes rays which have the same origin and nearly the same directions to intersect the same object at nearly the same location. This coherence category has been developed exclusively to accelerate ray tracing algorithm. It is used by [12] to efficiently compute the intersection between a ray and a parametric surface patch. For consecutive pixels in which the same patch is seen, solution of the previous pixel can be used as an initial value for quasi-Newton iteration. Another utilisation of ray coherence specific to polygonal objects is found in [10]. By making a slight modification to the formulation for refraction, it is possible to maintain coherence and regular structure of first generation rays even after reflection and refractions. The most recent paper utilising this coherence property is found in [14]. It is used in constant, with number of object, ray tracing algorithms.

3. RESEARCH METHODOLOGY

The approach taken to compare the performance of rendering algorithms that has been used for nearly all algorithms described so far relies on measuring rendering time with a specific and limit set of realistic objects or CSG models. The complexity of realistic objects and CSG models is known in the construction process. The result is then used to calculate the gain factor and finally used in comparison with the reference algorithm in determining the improvement of algorithms with and without exploiting coherences.

Unfortunately, the performance of a display algorithm for a particular image of a CSG model depends heavily on complexity factors intrinsic to the model and view-dependent factors. The exact relations between these factors and the performances of the display algorithms, might be established by a theoretical analysis of the complexity of the algorithms, or by a statistical analysis of measurements involving hundreds of realistic test models. This would result in expressions for the complexity of the algorithms in terms of the factors. Unfortunately, the algorithms are non-trivial for a detailed theoretical analysis, and hundreds of realistic test models are not available. Moreover, even if the mathematical expressions were obtained, it would be doubtful whether these would give much insight into the practical behaviour of the algorithms.

From our point of view, a good approach to compare the performance of the algorithms should not rely only on a limit set of CSG models. Because the designer of rendering algorithms might have a bias against a whole set of available models and make the comparison results uncertainty. We believe that the relations between rendering time and complexity factors can be established without utilising realistic models. In addition, the complexity can be approximated by use of some form of artificial models.

In general, the performance of rendering algorithms is mainly affected by the following complexity factors:

-**X** : Complexity of primitives in CSG models: number of faces, vertices and edges,
-**N** : Number of primitives in CSG models,
-**C** : Class of primitives in CSG models: quadric, quartic, halfspaces.

Therefore, in comparing the performance of rendering algorithms, we need to find the relationship between the rendering time, R, with these factors. This relationship can be written in the mathematical form:

$$R \propto f(N, C, X).$$

In the next section, we describe more details about the proposed technique to study the benefits of coherence exploitation with respect to the above factors.

4. PROPOSED TECHNIQUE

In order to study the benefit of coherent exploitation in rendering algorithms, we have two alternatives. The first compares the rendering time of the algorithm with and without coherent exploitation. The second compares the rendering time from two set of CSG models; the first set with high coherence and the second without coherence but the same geometric complexities (stated above). In other words, we can say that the first alternative tries to modify the nature of rendering algorithms, whereas the second tries to modify the inputs to rendering algorithms. It is obvious that the first alternative is nontrivial. Because coherent exploitations are inherent properties of rendering algorithms. This means that any methods which try to disable coherence exploitation may violate the behaviour of the algorithm. In contrast, the second alternative has nothing to do with rendering algorithms. Therefore, we propose the latter method.

From now on, we call CSG models to be used in our proposed technique as *artificial models* in contrast to realistic models. It is obvious to us that realistic models is not useful to form the relations described in the previous section due to the difficulty to control variables and complexities. Following, we describe our proposed technique. First of all, to make sure that models in high and low coherence above have the same geometric complexities, we use "primitive perturbation" to destroy coherence in artificial models. The definition of primitive perturbation is: "*Primitive perturbation destroys the coherence of artificial models by translation and rotation of the selected primitive node in the CSG tree of the artificial model. The overall process must not change geometric properties of the CSG models; i.e. increase number of primitives, faces, or edges.*" The method destroys coherence in artificial models from the fact that the final models do not have the similarity within their structure anymore.

In generating artificial models for rendering algorithm, we start from an atom which is a CSG model constructed from group of **N** primitives of type **C** with known complexity **X**. The artificial model is constructed by uniformly distribute all **P** atoms in three dimensional space. The CSG tree of artificial model is union of all **P** atoms. The rendering time for this high coherence artificial model is measured at this stage. Then, primitive perturbation is operated to all atoms $p \in P$ in the CSG tree of artificial model. To make sure that we do not have any bias against any atom, we generate both translational and rotational operation in random way for each atom p ($p \in P$). It is noted here that whereas it is trivial to generate random translational, care must be taken for the case of random rotation. Because the generation of random rotations is related to random orientation. Therefore, the method of generating random unit vectors on the surface of sphere must be used in this case. The artificial model in this stage is in a low coherence.

By repeating the above process in different configurations of **N**, **T**, and **X**. The proposed technique can describe the characteristic curves of rendering time for an algorithm for our artificial models in the following forms:

"Given **P** atom constructed from **N** primitives of the known type **T** (cuboid, .., sphere) and complexity, if the artificial model is constructed by uniformly distribute all **P** atoms in three dimensional space. The characteristic curves of rendering time of algorithm A when **P** is varied with and without coherence exploitation are...",

5. EXPERIMENTS

To evaluate the benefits of coherence exploitation in rendering algorithms, we tested our models with 2 standard rendering algorithms, that are CSG ray-casting (simple bound, s-bound and octree versions) and CSG z-buffering, on the DEC Alpha workstation under the VMS operating system. Currently, we are conducting experiments on the artificial models with variations in geometric complexities and number of primitives with and without coherence. The rendering time for each configuration of artificial CSG models is measured by using the DEC Performance Coverage Analyser (PCA). To make sure that the rendering time is correct, we try to eliminate all factors which do not have an effect on the rendering time from the PCA result; i.e. time spent on system space and page-faults. The experimental results will be presented in a separate paper.

6. SUMMARY AND CONCLUSION

This paper has described a technique for evaluating the benefits of coherence exploitation in CSG rendering algorithms. The proposed technique is used to establish relations between the following properties in connection and the rendering time and the coherence exploitation:

-**X** : Complexity of primitives in CSG models: number of faces, vertices and edges,

-**N** : Number of primitives in CSG models,

-**C** : Class of primitives in CSG models: quadric, quartic, halfspaces.

7. ACKNOWLEDGEMENTS

I am grateful to Dr. Anthony Barrett for reading this paper and to Prof. Alan Middleditch for initiating and giving a brief idea of the evaluation method and a permission to use his solid modeller. In addition, I am very appreciate for the valuable technical support from Dr. Gerry Agnew of the CGMD, Brunel University. I gratefully acknowledge the Thai Governmental Scholarship for supporting the research.

8. REFERENCES

1. P. R. Atherton, "A Scanline Hidden Surface Removal Procedure for Constructive Solid Geometry", Computer Graphics, Vol. 17, No. 3, pp. 73-82, 1983.
2. W. F. Bronsvoort, J. J. van Wijk, and F. W. Jansen, "Two Methods for Improving the Efficiency of Ray Casting in Solid Modelling", Computer-Aided Design, Vol. 16, No. 1, pp. 51-55, 1984.
3. W. F. Bronsvoort, "Techniques for Reducing Boolean Evaluation Time in CSG Scan-Line Algorithm", Computer-Aided Design, Vol. 18, No. 10, pp. 533-538, 1986.
4. S. Cameron, "Efficient Bounds in Constructive Solid Geometry", IEEE Computer Graphics and Applications, Vol. 11, No. 3, pp. 68-74, 1991.
5. G. A. Crocker, "Screen-Area Coherence for Interactive Scanlıne Display Algorithm", IEEE Computer Graphics and Applications, Vol. 7, No. 9, pp. 10-17, 1987.
6. A. Fujimoto, T. Tanaka, and K. Iwata, "ARTS: Accelerated Ray-Tracing System", IEEE Computer Graphics and Applications, Vol. 6, No. 4, pp. 16-26, 1986.
7. A. S. Glassner, "Space Subdivision for Fast Ray Tracing", IEEE Computer Graphics and Applications, Vol. 4, No. 10, pp. 15-22, 1984.
8. N. Green, M. Kass, and G. Miller, "Hierachical Z-buffer visibility", Computer Graphics Proceedings, Annual conference series, pp. 231-238, 1993.
9. J. G. Griffiths, "A Depth-Coherence Scanline Algorithm for Displaying Curved Surfaces", Computer-Aided Design, Vol. 16, No. 2, pp. 91-101, 1984.
10. P. S. Heckbert and P. Hanrahan, "Beam Tracing Polygonal Objects", Computer Graphics, Vol. 18, No. 3, pp. 119-127.
11. F. W. Jansen, "Depth-order point classification techniques for CSG display algorithms", ACM Transactions on Graphics, Vol. 10, No. 1, pp. 40-70, 1991.
12. K. I. Joy and M. N. Bhetanabhotia, "Ray Tracing Parametric Surface Patches Utilising Numerical Techniques and Ray Coherence", Computer Graphics, Vol. 20, No. 8, pp. 279-285.
13. R. van Kleij and W. F. Bronsvoort, "Performance of Scanline Algorithms for Direct Display of CSG", The Visual Computer, Vol. 10, pp. 317-329, 1994.
14. M. Ohta and M. Maekawa, "Ray Coherence and Constant Time Ray Tracing", Computer Graphics 1987 (Proceedings of CG International '87), Editor: T. L. Kunii, pp. 303-314, Springer-Verlag, 1987.
15. X. Pueyo, and J. C. Mendoza, "A New Scanline Algorithm for the Rendering of CSG trees", Proceeding of Eurographics'87, Editor: G. Marechal, pp. 347-362, Elsevier Science, Amsterdam, 1987.
16. J. R. Rossignac and A. A. G. Requicha, "Depth Buffering display Techniques for CSG", IEEE Computer Graphics and Application, Vol. 6, No. 9, pp. 29-39, 1986.
17. J. R. Rossignac and H. B. Voelker, "Active Zones in CSG for Accelerating Boundary Evaluation, Redundancy Elimination, Interference Detection, and Shading Algorithm", ACM Transactions on Graphics, Vol. 8, No. 1, pp. 51-86, 1989.
18. S. D. Roth, "Ray Casting for Modelling Solids", Computer Graphics and Image Processing, Vol. 18, No. 2, pp. 109-144, 1982.
19. D. Salesin, J. Stolfi, "Rendering CSG Models with a ZZ-Buffer", Computer Graphics (SIGGRAPH), Vol 24, No 4, pp 67-76, 1990.
20. K. H. Sears, and A. E. Middleditch, "Set Theoretic Volume Model Evaluation and Picture-Plane Coherence", IEEE Computer Graphics and Application, Vol. 4, No. 4, pp. 41-46, 1984.
21. J. Woodwark, "Generating Wireframes from Set-Theoretic Solid Models by Spatial Division", Computer-Aided Design, Vol. 18, No. 6, pp. 307-315, 1986.

SMOOTH SURFACE INTERPOLATION OF IRREGULAR MESHES

Xuefu Wang and **Fuhua(Frank) Cheng**
Department of Computer Science
University of Kentucky
Lexington, KY40506, USA

ABSTRACT

A new method for interpolation of irregular meshes with arbitrary topological type is presented. The new approach first restructures the given mesh to obtain a new mesh suitable for network curves construction, and then fills all the n-sided holes framed by the network curves with n-sided Varady patches. The smoothness of the resulting surface is achieved through a global energy minimization process. The resulting surface is C 1 continuous. Advantages of the new method include: (1) fewer-patches in the resulting surface than methods that require composition of triangular or quadrilateral patches to form an n-sided surface; (2) low degree parametric representation of the resulting surface (only bicubic patches are needed in the construction of the final surface); (3) overall smoothness and more uniform distribution of curvature because of the global energy minimization process.

KEYWORDS

Interpolation, irregular mesh, strain energy, energy minimization, Varady patch

1. INTRODUCTION

Given an irregular mesh, this paper deals with the problem of constructing a smooth parametric surface that interpolates the vertices (position and/or normal vectors) of the mesh. By an irregular mesh, we mean a mesh that is not homeomorphic to a rectangular grid.

Subdivision surfaces have been used to perform such a task. *Sabin-Doo* subdivision surfaces [2-3] are used in Nasri's attempt [9]. For a given irregular mesh, a new mesh with the same topology (connectivity) is constructed first. The patch centroids of the new mesh coincide with the corresponding vertices of the original mesh. The new mesh is then repeatedly subdivided to generate a Sabin-Doo subdivision surface. Since a Sabin-Doo subdivision surface passes through the centroids of its patches, the subdivision surface of the new mesh, consequently, interpolates the vertices of the original mesh. However, this method can not interpolate the normal vectors of the mesh vertices. Halstead etc [5] overcome this shortage by using *Catmull-Clark* subdivision surfaces [1] in their approach. The basic idea is to provide the new mesh with more freedom (more vertices than the original one). Like any subdivision based algorithm, these methods do not admit analytical form for the resulting surface.

Parametric spline surfaces have also been used to solve this problem. Fasshauer and Schumaker [4] use parametric patch complexes whose components are drawn from linear spaces of polynomial splines to interpolate the mesh. The polynomial splines are defined on a triangulation of a parametric domain in R^2. This is basically a 2D method. It can not model shapes of arbitrary topology and does not have the ability to interpolate normal vectors.

In this paper, we present a new approach for the above problem. The algorithm works conceptually in the following fashion: Given an irregular 3D polyhedral mesh of arbitrary topology, it first restructures the mesh to generate a new mesh. This restructuring process increases the number of edges, vertices and faces of the given mesh, but it imposes a simpler structure on the resulted mesh. In the new mesh, all the interior vertices have incidence degree of only four. The next step is to construct a network of curves through the vertices of the new mesh. The third step is to fill in all the n-sided holes framed by the network of curves to form a C^1 continuous surface whose strain energy is minimum. The advantages of this new technique over existing schemes include generating fewer patches, lower surface degree, and minimal energy.

The remaining part of the paper is organized as follows: In Section 2, we present the refinement techniques for a given irregular mesh. In Section 3, the procedure to construct a network of curves through the vertices of a restructured mesh is shown. In Section 4, we present the algorithm that fills the n-sided holes framed by the network of curves. In Section 5, we show implementation details and some examples. Concluding remarks are given in Section 6.

2. Refinement

Given an irregular mesh M_0, we first construct a new mesh M_1 with the same topology as the original one M_0, where the centroid of each *vertex face* of M_1's subdivision mesh satisfy the following condition:

$$O - V = (V - V')/3$$

where V is a vertex of the original mesh M_0, V' is the corresponding vertex in the new mesh M_1, and O is the centroid of the vertex face in M_1's subdivision mesh which corresponds to V'. M_1 can be obtained by solving a system of linear equations for all vertices.

After this step, a Loop subdivision technique [7] is performed on M_1 as follows. For each face of the new mesh M_1, a new vertex is generated for each of its vertices. If V_i is a vertex of the patch $F = V_0 V_1 \cdots V_{n-1}$, the new vertex for V_i is generated as follows:

$$V_i' = V_i/2 + O/4 + V_{i-1}/8 + V_{i+1}/8$$

All the subscripts are taken modulo n. The new vertices are then connected to form three types of new faces: *vertex-faces*, *edge-faces* and *polygon-faces* which correspond to M_1's vertices, edges and faces, respectively. The connection scheme is shown in Figure 1.

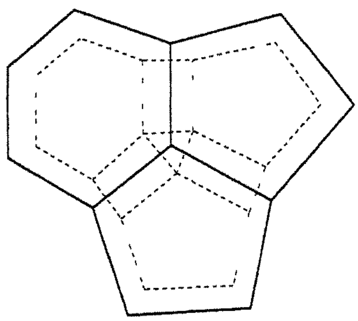


Figure 1. Subdivision and connection scheme.

The newly formed mesh is called M_2. This new mesh M_2 is closer to the original mesh than M_1 and, consequently, it is easier for us to interpolate the original vertices.

3. Netwrok Curve Construction

The next step is to construct a network of curves to interpolate the vertices of M_2. Note that the new mesh M_2 has the following property: each inner vertex has an incidence degree of four, and each boundary vertex has an incidence degree of two or three.

For each edge of mesh M_2, we extend it in both directions to construct a path. The rule for the extension is: among the free edges incident to the current end vertex of the path, choose an edge that is not not in the same face as the current end edge in the new mesh M_2. The extension is performed until either a closed path is formed or the boundary of the mesh is reached. For each path, we then construct a *Cubic Hermite Spline curve* to interpolate its vertices. The resulting curves form a network that frames a set of n-sided holes. Our goal is to fill in these holes with n-sided surface patches that would not form a smooth surface (C^1 continuous), but also interpolates the vertices (and/or normal vectors) of the original mesh M_0.

4. Filling

For each n-sided hole framed by the network curves, we construct an n-sided Varady patch to fill it. To define an n-sided Varady patch, one needs to know the vertices of the patch, left and right tangent vectors at each vertex, and the patch's twist vector at each vertex [16-17]. The patch is defined on a regular n-gon with unit sides, using an overlapping scheme.

Let D be a regular n-gon with unit sides whose vertices are $D_0, D_1, \ldots, D_{n-1}$. Given n three-dimensional points V_i, $i=0,1,\ldots,n-1$, and 3 vectors L_i, R_i, and T_i for each V_i as the left and right tangent vectors and the twist vector, an n-sided patch can be defined as follows:

$$p(u,v) = Q + \sum_{i=0}^{n-1} p_i(u_i,v_i), \quad 0 \le u_i, v_i \le 1, \quad (u,v) \in D$$

where Q is a three-dimensional point, called the *origin* of the patch, and $p_i(u_i,v_i)$ are *vertex overlap patches* (or, simply, {\m vertex patches}) defined as follows:

$$p_i(u_i,v_i) = (V_i - Q)f(u_i)f(v_i) + R_i g(u_i)f(v_i)$$
$$+ L_i f(u_i)g(v_i) + T_i g(u_i)g(v_i)$$

where $f(t)$ and $g(t)$ are *Hermite basis functions* and the domain of each vertex overlap patch is a unit square.

Given a point (u,v) in D, to get the corresponding point on the patch, one needs to compute the corresponding local parameters (u_i,v_i) and the value of $p_i(u_i,v_i)$ for each i, and then compute the sum of all the vertex overlap patches. The origin of the n-sided patch, Q, is introduced to ensure invariance under affine transformations. It also provides an extra degree of freedom for one to set the midpoint of the patch. Q does not have any effect on the patch when $n=4$ since, in this case, $\sum_{i=0}^{3} f(u_i)f(v_i)=1$. To see how Q affects the shape of non-four-sided patches, refer to Varady's original work [16-17].

Since the tangent vectors at each vertex are available to us from the network curves constructed in the previous section, we only need to calculate a twist vector for each vertex and the local origin for each non-four-sided patch. No efficient solution for estimating the twist vectors and local origin of an n-sided patch are currently available. Varady suggested using local heuristics method. We present a global method for solving such a problem by minimizing the entire surface's strain energy. According to the energy calculation method for single n-side patch, we know that in our case, the energy depends on the twist vectors at the vertices and the patch's local origin.

By calculating each single patch's energy, we could represent the entire surface's energy in terms of twist vectors at the vertices, and local origins of non-four-sided patches. It has the following form. Let T be the set of all twist vectors and Q be the set of all local origins.

$$T=(T_1,T_2,\cdots,T_m); \quad Q=(Q_1,Q_2,\cdots,Q_k)$$

where m is the number of vertices of the curve network and k is the number of non-four sided patches, and let $S=(T_x,Q_x,T_y,Q_y,T_z,Q_z)$. After adding up the energies for all patches, we have the following energy representations:

$$E=const+2*S*B+S*A*S_t$$

where B is a $3(k+m)$ vector, and A is a $(3(k+m))\times 3(k+m))$ matrix.

We need to minimize the above energy with the following interpolation constraints:

- Position Constraint. Each vertex patch must pass through the corresponding vertex of the original mesh M_0. Suppose the vertex patch has parametric form $S_i(u,v)$, the corresponding vertex is V_i, then the constraints is $S_i(0.5,0.5)=V_i$, According to the definition of n-side patch, we know that this is a linear constraint. By grouping all the constraints together, we get a system of linear constraints.

- Normal Constraint. Each vertex patch must meet the normal vector requirement, that is $S_{iu} \cdot N=0$ and $S_{iv} \cdot N=0$, so at most for each vertex patch, the normal vector constraint can turn into 2 linear constraints.

So, there are at most $3l$ (l is the number of vertex patches of M_2, the same as the number of vertices of M_0) linear constraints for the above minimization, there are $k+m$ unknowns (k,m are the numbers of vertices and non-four-sided patches of the mesh M_2, respectively), and $k > 3l$ (one can easily prove this by looking into the Loop subdivision process). So this is a linearly constrained minimization problem with the number of constraints less than the number of free variables. It must have solutions, and can be solved quickly using the method given below.

Since the constraint is an under-determined linear equation system, which can be represented by the equation $M*S=D$ where M is the constraint matrix and S is the vector of all unknowns, any solution for this constraint equation system can be represented as S_0+C*R, where S_0 is any solution to the constraint equation, while R is a matrix whose row vectors are the basis vectors of the null space of matrix M, and C is a vector to be determined. Substituting the above solution representation into the entire surface's energy form, we get:

$$E(s)=const+2C*R*B+2C*R*A_0*S_0^t+C*(R*A*R^t)*C^t$$

Since matrix $R*A*R_t$ is a positive definite matrix (because A is a positive definite matrix), the C can be solved by solving the following equation:

$$C*(R*A*R^t)+R*B+R*A*S_0^t=0$$

So, we can calculate the twist vectors and local origin for every non-four-sided patch, hence the entire surface can be constructed with all the interpolation constraints satisfied.

5. Implementation

The above system has been implemented using C and X-windows on HPUX 9.05 on an HP 735 machine using a SUN Sparc 20 machine as the display device. The software package used for solving a system of linear equations is the *NAG Fortran Library*. A number of test cases have been investigated. Two of them are shown in Figures 2-3.

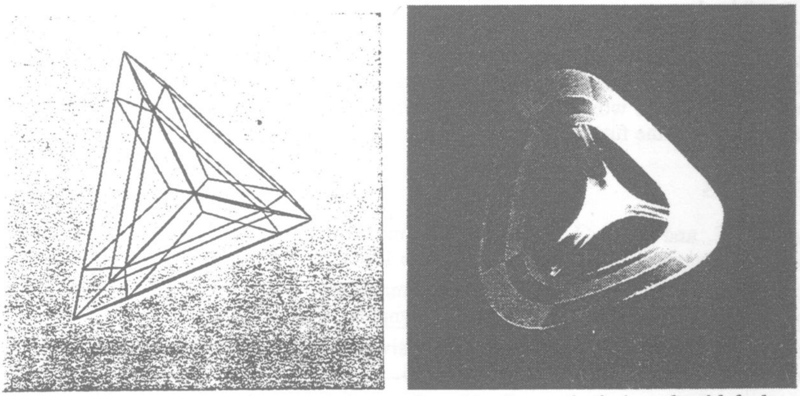


Figure 2. Wireframe (a) and interpolating surface (b) of a tetrahedral mesh with holes.

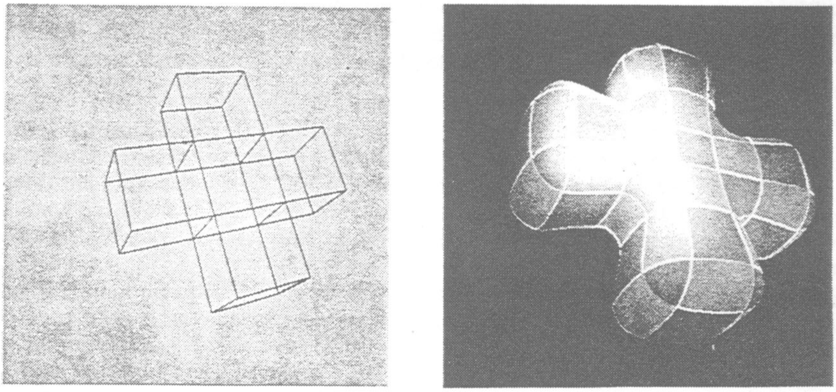


Figure 3. Wireframe (a) and interpolating surface (b) of a concave-convex polyhedral mesh.

6. Conclusions

We have presented an new method for interpolation of of irregular meshes with arbitrary topological type. The new method first first restructures the given mesh to obtain a new mesh suitable for network curves construction, and then fills all the n-sided holes framed by the network curves with n-sided Varady patches. The smoothness of the resulting surface is achieved through a global energy minimization process. The resulting surface is $C1$ continuous. Several new techniques have been introduced, these include restructuring techniques for a given mesh, network curves construction technique, energy formulation technique for n-sided Varady patches, and energy computation technique for Varady surfaces.

Advantages of the new method include: (1) fewer patches in the resulting surface than methods that require composition of triangular or quadrilateral patches to form an n-sided surface; (2) low degree parametric representation of the resulting surface (only bicubic patches are needed in the construction of the final surface); (3) overall smoothness and more uniform distribution of curvature because of the global energy minimization process.

Actually, the technique presented here can be used for blending and smoothing of irregular meshes as well. Work in that direction is currently undergoing and initial results are very promising [18].

A possible shortcoming of our approach is the computation time required in the energy minimization process. The time needed to optimize a system of linear equations by the NAG subroutine is the dominating factor of the entire process. For a system of 1,000×1,000 linear equations, it takes NAG about 38 seconds to carry out the task. For a system of 2,900×2,900 linear equations, it takes NAG about six minutes to finish the work. Although tolerable, it will be a major goal for us to reduce the time needed for this part in the future.

REFERENCES

1. Catmull, E. and Clark, J., "Recursively Generated B-Spline Surfaces on Arbitrary Topological Meshes", *Computer Aided Geometric Design* 10(6), pp. 350-355, 1978.
2. Doo, D., "A subdivision algorithm for smmothing down irregularly shaped polyhedrons", in Proceedings of Interactive Techniques in Computer Aided Design, pp. 157-165, Bologona, 1978.
3. Doo, D. and Sabin, M., "Behaviour of recursive devision surfaces near extraordinary points", *Computer Aided Design* 10(6), pp. 356-360, 1966.
4. Fasshauer, G.E. and Schumaker, L.L., "Minimal energy surfaces using parametric splines", *Computer Aided Geometric Design* 13, pp. 45-79, 1996.

5. Halstead, M., Derose, T. and Kass, M., "Efficient, Fair Interpolation Using Catmull-Clark Surfaces", *Computer Graphics* 27(2), pp. 35-44, 1993.

6. Kallay, M. and Ravani, B., "Optimal twist vectors as a tool for interpolating a network of curves with a minimum energy surface", *Computer Aided Geometric Design* 7, pp. 465-473, 1990.

7. Loop, C., "Smooth Spline Surfaces Over Irregular Meshes", *Computer Graphics* 28(2), pp. 303-310, 1994.

8. Lounsbery, M , Mann S. and Derose, T., "Parametric surface interpolation", *IEEE Computer Graphics & Applications* 12(5), pp. 45-52, 1992.

9. Nasri, A., "Polyhedral subdivision methods for free-form surfaces", *ACM Transactions on Graphics* 6(1), pp. 29-73, 1987.

10. Nasri, A., "Boundary-corner control in recursive-subdivision surfaces", *Computer Aided Design* 23(6), pp.405-410, 1991.

11. Nasri, A., "Surface interpolation on irregular networks with normal conditions", *Computer Aided Geometric Design* 8, pp. 89-96, 1991.

12 Peters, J., "Smooth free form surfaces over irregular meshes generalizing quadratic splines", *Computer Aided Geometric Design* 10, pp. 347-361, 1993.

13. Peters, J., "Smooth Interpolation of a Mesh of Curves", *Constructive Approximation* 7, pp. 221-246, 1991.

14. Peters, J., "Smoothing Polyhedra Made Easy", *ACM Transactions on Graphics* 14(2), pp. 162-170, 1995.

15. Peters, J., "C^1-surface splines", *SIAM J. NUMER. ANAL* 32(2), pp. 645-666, 1995.

16 Varady, T., "Overlap Patches: a new scheme for interpolating curve networks with n-sided regions", *Computer Aided Geometric Design* 7, pp. 7-27, 1990.

17. Varady, T., Survey and New results in n-sided patch generation, The Mathematics of Surfaces II, R. Martin ed., Oxford Univ., pp. 203-235, 1987.

18. Wang, X., Cheng, F. and Barsky, B.A., "Blending, Smoothing and Interpolation of Irregular Meshes Using N-Sided Varady Patches", in preparation.

NEW CURVES AND SURFACES COMPARING FAVOURABLY WITH BEZIER CURVES AND SURFACES IN COMPUTER - AIDED GEOMETRY DESIGN

J.D. CAO
Department of Mathematics
Fudan University
Shanghai, P.R. of China

ABSTRACT

Bézier curves and surfaces are useful in the aircraft and automotive industries, where shape is all important. The author constructs new curves and surfaces giving a much better closeness of fit, this paper describes a computer method for shape design in concrete and practical cases.

KEYWORDS

Bézier curves and surfaces, Shape design, New curves and surfaces, Properties, Numerical results

1. INTRODUCTION

Bézier (at Renault company, France) introduced the famous curves and surfaces in CAGD using basic functions of Bernstein polynomials. Bézier curves and surfaces are useful in the aircraft and automotive industries, where shape is all important. Butzer posed a problem which he said: "··· the problem has probably been part of the mathematical folklore for some generation ···". Butzer's problem: Can one construct discrete positive linear algebraic polynomials of degree n having best possible rate of approximation? The Bernstein polynomials only have slow rate of approximation. The author and H.H. Gonska constructed new polynomials and solved Butzer's problem. By the basic functions of new polynomials we constructed new curves and surfaces. We presented two talks at the Society for Industrial and Applied Mathematics (SIAM) of USA. These curves and surfaces possess a number of properties shared by Bézier curves, but give a much better closeness of fit. We cite applications in computer - aided design, analysis, and manufacturing. In 1994, we had researched the new curves and surfaces for theoretical cases, now we give numerical results for shape design in concrete and practical cases.

2. BUTZER'S PROBLEM

Let $h \in C[a,b]$, $\|h\| = \|h\|_{C[a,b]} = \max_{a \le t \le b} |h(t)|$. Let $h(x) \in C[0,1]$, Bernstein polynomial of degree n is defined by

$B_n(h(x),t) = \sum_{i=0}^{n} h\left(\frac{i}{n}\right) J_{n,i}(t),$ where $J_{n,i}(t) = \binom{n}{i} t^i (1-t)^{n-i}$, $J_{n,i}(t)$ are called basic functions of Bernstein.

Let $\bar{P}_i (i = 0, 1, \cdots, n)$ be vectors of vertices of a characteristic polygon of space. Bézier curve is defined by

$$\bar{P}(t) \overset{def}{=} \sum_{i=0}^{n} J_{n,i}(t)\bar{P}_i, \qquad 0 \le t \le 1.$$

Let $\bar{P}_{i,j}$ (i = 0, 1, ...n; j = 0, 1, ...m) be vectors of space, the Bézier surface of m x n degree is defined by

$$\text{Let } \bar{P}(u,v) \overset{def}{=} \sum_{i=0}^{n}\sum_{j=0}^{m} \bar{P}_{i,j} J_{n,i}(u) J_{m,j}(v), \quad \begin{matrix} 0 \le u \le 1, \\ 0 \le v \le 1. \end{matrix}$$

Director Butzer asked the following problem (an old problem in approximation theory) [4] [5]:

Can one construct a triangular matrix of distinct nodes $\{t_{r,n}\}_{r=0}^{n}$ ($n = 0, 1, 2, \cdots, -1 \le t_{r,n} \le 1$) and a triangular matrix of positive fundamental functions $\{\varphi_{r,n}\}_{r=0}^{n}$ define on [-1, 1], such that the linear summator operators $L_n(f,t) \overset{def}{=} \sum_{r=0}^{n} f(t_{r,n})\varphi_{r,n}(t)$, $(f \in C[-1,1])$ are algebraic polynomials of degree n and satisfy $\|L_n f - f\|_{C[-1,1]} = 0(n^{-\alpha})$, provided $f \in Lip_2(\alpha, C)$ and $(0 < \alpha \le 2)$?

Recall that under the same hypotheses the Bernstein polynomials of f approximate f only with order $0\left(n^{-\frac{\alpha}{2}}\right)$.

In the above, $Lip_2(\alpha, C)$ denotes the set of all $f \in C[-1, 1]$ for which the second order modulus of smoothness satisfies $\omega_2(f, \delta) = 0(\delta^{\alpha}), \delta \to +0$.

In [6] [7] [8] I and Gonska had given various solutions of Butzer's problem, we had presented talks at Texas conference and Memphis conference of USA.

We describe a computer method for shape design in concrete and practical cases. Let

$$K_{3n-3}(v) \overset{def}{=} \frac{10}{n(11n^4 + 5n^2 + 4)} \left(\frac{\sin\left(\frac{nv}{2}\right)}{\sin\left(\frac{v}{2}\right)} \right)^6$$

$$= \frac{1}{2} + \sum_{i=1}^{3n-3} \rho_{i,3n-3} \cos iv, \qquad \text{(see [6])},$$

here $\rho_{1,3n-3} = \dfrac{11n^4 - 5n^2 - 6}{11n^4 + 5n^2 + 4}$, $\rho_{2,3n-3} = \dfrac{11n^4 - 35n^2 + 24}{11n^4 + 5n^2 + 4}$.

Let $T_i(t) = \cos(i \arccos t)$ be the ith Chebyshev polynomial, and Chebyshev nodes of Nth degree $t_{r,N} = \cos\theta_{r,N}$, $\theta_{r,N} = \frac{2r-1}{2N}\pi$, $(1 \le r \le N)$, we construct new polynomial (see [6])

$$\Lambda_{3n-3}(f,t) \stackrel{\text{def}}{=} t\sum_{r=1}^{3n-2} f(t_{r,3n-2})A_{r,3n-2}(t), \tag{1}$$

$$A_{r,3n-2}(t) \stackrel{\text{def}}{=} \frac{1}{3n-2}\left\{1+2\sum_{i=1}^{3n-3}\rho_{i,3n-3}T_i\left(t_{r,3n-2}\right)T_i(t)\right\}$$

$$= \frac{1}{3n-2}\left\{K_{3n-3}\left(\theta-\theta_{r,3n-2}\right)+K_{3n-3}\left(\theta+\theta_{r,3n-2}\right)\right\}, (\theta = \arccos t), \tag{2}$$

let [a] be integral part of number a, we have $3\left(\left[\frac{n}{3}\right]+1\right)-3 \le 3\left[\frac{n}{3}\right] \le n$, the $\Lambda_{3\left[\frac{n}{3}\right]}(f,x)$ (algebraic polynomial of degree $\le n$) give one solution of Butzer's problem.

3. NEW CURVE AND ITS PROPERTIES

Let $\vec{P}_r (r = 1, 2, \cdots, 3n\text{-}2)$ be vectors of vertices of a characteristic polygon of space. We construct new curve

$$\vec{\Lambda}_{3n-3}(t) \stackrel{\text{def}}{=} \sum_{r=1}^{3n-2} A_{r,3n-2}(t)\vec{P}_r, \quad -1 \le t \le 1, \tag{3}$$

Let n be natural number, the curve $\vec{\Lambda}_{3n-3}(t)$ has the following properties: (1) (Consistency). If all vertices of characteristic polygon are congruent to a point, then the generated curve $\vec{\Lambda}_{3n-3}(t)$ is congruent to a point. (2) (Preservation of straight line). If characteristic polygon reduces to a straight line, then generated curve $\vec{\Lambda}_{3n-3}(t)$ reduces to a straight line. (3) Property of convex hull. (4) Geometric invariance. (5) $\{A_{r,3n-2}(t), 1 \le r \le 3n-2\}$ is system of linear independent functions on [-1,1]. (6) Property of approximation.

Th requirement having continuous 1st derivative is sufficient for shape design of components of general machine. The requirement having continuous 2nd derivative is necessary for shape design of airplane and ship, we prove that (1) If $f(t) \in C[-1,1]$, then $\lim_{n\to\infty} \|\Lambda_{3n-3}(f,t) - f(t)\| = 0$. (2) If f(t) has kth continuous derivative $f^{(k)}(t)$ for (k=1, 2) on [-1,1], then $\lim_{n\to\infty}\left\| \frac{d^k}{dt^k}\Lambda_{3n-3}(f,t)-f^{(k)}(t)=0 \right\|$, (see [9]).

4. NEW SURFACE

Let $\vec{P}_{r,p}$ $(r = 1, 2, \cdots, 3n\text{-}2;\ p = 1,2,\ldots 3m\text{-}2)$ be vectors of space, we construct new surface of degree $(3m-3)\times(3n-3),\begin{pmatrix}-1\le u\le 1\\ -1\le v\le 1\end{pmatrix}$

$$\vec{U}_{3n-3,3m-3}(u,v) \overset{\text{def}}{=} \sum_{r=1}^{3n-2}\sum_{P=1}^{3m-2} A_{r,3n-2}(u)\cdot A_{P,3m-2}(v)\vec{P}_{r,P}, \tag{4}$$

$$A_{r,3n-2}(u) = \frac{1}{3n-2}\left\{K_{3n-3}(\theta-\theta_{r,3n-2}) + K_{3n-3}(\theta+\theta_{r,3n-2})\right\}, (\theta = \text{arc } \cos u),$$

$$A_{p,3m-2}(v) = \frac{1}{3m-2}\left\{K_{3m-3}(\theta-\bar{\theta}_{p,3m-2}) + K_{3m-3}(\theta+\bar{\theta}_{p,3m-2})\right\}, (\theta = \text{arc } \cos v),$$

$$\theta_{r,3n-2} = \frac{2r-1}{2(3n-2)}\pi,\ \bar{\theta}_{p,3m-2} = \frac{2p-1}{2(3m-2)}\pi.$$

Let n and m be natural numbers, the new surface has the following properties: (1) Property of convex hull, (2) Geometric invariance, (3) Property of approximation.

5. NUMERICAL RESULTS

If $n = 2$, then

$$\theta_r = \theta_{r,4} = \frac{(2r-1)}{8}\pi(1\le r\le 4),\ \bar{\theta}_p = \bar{\theta}_{p,4} = \frac{(2p-1)}{8}\pi,\ (1\le p\le 4),$$

$$A_{r,4}(u) = \frac{1}{4}\left\{\left(1-\frac{3}{5}\cos 2\theta_r\right) + \left(\frac{3}{2}\cos\theta_r - \frac{3}{10}\cos 3\theta_r\right)u + \frac{6}{5}\cos 2\theta_r\cdot u^2 + \frac{2}{5}\cos 3\theta_r\cdot u^3\right\},$$

$$A_{p,4}(v) = \frac{1}{4}\left\{\left(1-\frac{3}{5}\cos 2\bar{\theta}_p\right) + \left(\frac{3}{2}\cos\bar{\theta}_p - \frac{3}{10}\cos 3\bar{\theta}_p\right)v + \frac{6}{5}\cos 2\bar{\theta}_p\cdot v^2 \frac{2}{5}\cos 3\bar{\theta}_p\cdot v^3\right\},$$

By (3) and (4) we get parametric cubic curve and Bi-parametric cubic surface, see talk of author at Chamonix conference [10].

6. REFERENCES

1. Su, B.Q. and Liu, D.Y.; Computational Geometry (in Chinese) Press of Sci and Tech of Shanghai, 1981.
2. Farin, G.; Curves and Surfaces for Computer Aided Geometric Design, A Practical Guide, Academic Press, Inc., 1988.

3. Su, B.Q. and Hua, X.J.; Course of Applied Geometry (in Chinese), Press of Fudan Univ., 1990.
4. Butzer, P.L. and Stens, R.L. and Wehrens, M.; Approximation Theory and Functional Analysis (edited by Prolla, J.) (Math. Studies vol.35), pp.71-102, North Holland, Amsterdam, 1979.
5. Butzer, P.L.; Functions, Series, Operators (edited by Nagy, B.Z. and Szabados, J.) (vol.1), pp.277-301, North Holland, Amsterdam, 1983.
6. Cao, J.D. and Gonska, H.H.; Approximation by Boolean sums of positive linear operators III: Estimates for some numerical approximation schemes, Numer. Funct. Anal. and Optimiz, 10 (7&8), pp.643-672, (1989).
7. Cao, J.D. and Gonska, H.H.; Approximation Theory VI (edited by Chui, C.K. and Schumaker, L.L. and Ward, J.D.), pp.117-120, Academic Press, New York, 1989.
8. Cao, A.D. and Gonska, H.H.; Approximation Theory (edited by Anastassiou, G.A.), pp.289-313, Marcel Dekker, Inc., 1992.
9. Cao, J.D. and Gonska, H.H. and Kacsó, D.P.; Simultaneous Approximation by Discretized Convolution - Type Operators, SM-DU-345, Duisburg Univ (Germany, pp. 1-24, (1996).
10. Cao, J.D. and Gonska, H.H. and Kacsó, D.P.; On some polynomial curves derived from trigonometric kernels, Proc of Chamonix 1996 (edited by Méhauté, A.Le. ··· etc.), Vanderbilt Univ., 1997.

Author/Session/Page Index

Author/Session/Page Index

ISBN: 962-85138-1-8 (1 Set)
ISBN: 962-85138-2-6 (Vol. 1)
ISBN: 962-85138-3-4 (Vol. 2)