A Dynamic Priority-based Approach to Concurrent Toolpath Planning for Multi-Material Layered Manufacturing

S.H. Choi and W.K. Zhu

Department of Industrial and Manufacturing Systems Engineering,
The University of Hong Kong,
Pokfulam Road, Hong Kong

Abstract

This paper presents an approach to concurrent toolpath planning for multi-material layered manufacturing (MMLM) to improve the fabrication efficiency of relatively complex prototypes. The approach is based on decoupled motion planning for multiple moving objects, in which the toolpaths of a set of tools are independently planned and then coordinated to deposit materials concurrently. Relative tool positions are monitored and potential tool collisions detected at a predefined rate. When a potential collision between a pair of tools is detected, a dynamic priority scheme is applied to assign motion priorities of tools. The traverse speeds of tools along the x-axis are compared, and a higher priority is assigned to the tool at a higher traverse speed. A tool with a higher priority continues to deposit material along its original path, while the one with a lower priority gives way by pausing at a suitable point until the potential collision is eliminated. Moreover, the deposition speeds of tools can be adjusted to suit different material properties and fabrication requirements. The proposed approach has been incorporated in a multi-material virtual prototyping (MMVP) system. Digital fabrication of prototypes shows that it can substantially shorten the fabrication time of relatively complex multi-material objects. The approach can be adapted for process control of MMLM when appropriate hardware becomes available. It is expected to benefit various applications, such as advanced product manufacturing and biomedical fabrication.

Key words:

Layered manufacturing; multi-materials; toolpath planning; multi-object motion planning; dynamic priority

1 Introduction

1.1 Layered Manufacturing

Layered manufacturing (LM), or rapid prototyping (RP), is an additive process that fabricates a physical prototype from a CAD model layer by layer, more rapidly than conventional manufacturing processes [1]. LM technology is now seen in a wide range of applications, such as product development, biomedical engineering, and architecture, etc. It offers huge potential to reduce or eliminate some stages of the traditional supply chain. The global market for LM products and services grew to an estimated USD1.183 billion in 2008, and the LM industry is expected to more than double in size by 2015, according to Wohlers Report 2009 [2].

LM processes can be roughly categorised as vector-based or raster-based. While a vector-based LM process drives a tool along a predefined path to deposit fabrication material, a raster-based process selectively generates specific contours out of an entire layer of material. Each of these LM processes offers distinctive traits for some specific types of prototypes [3].

Although LM can shorten prototyping cycles, the process is not as rapid as desired. Wohlers [2] pointed out that applications of LM are increasing, yet current LM systems are becoming unacceptably slow in respect of the increasing size and complexity of prototypes being made. Kochan [4] claimed that one of the main limitations of rapid prototyping was the low speed at which a part was fabricated. Bellini [5] presented that Fused Deposition Modelling (FDM) was fast enough for small parts of a few cubic inches, or those of tall, thin features, but it could be very time-consuming for parts with wide cross-sections. Hauser et al. [6] also pointed out that the methodology of LM was essentially a start-stop process because each layer was processed and deposited sequentially. The breaks in the build cycles, for example the positioning of hardware, often slowed down the build rate.

Some efforts have been devoted to enhancing the fabrication efficiency of LM. Sintermask Technologies [7] developed a machine with a Selective Mask Sintering (SMS) process capable of projecting infrared radiation through masks to sinter a whole layer of polyamide powder in ten seconds. Voxeljet [8] introduced a plastic powder binding system capable of building 400 cubic inches per hour. Hauser et al. [6]

developed a software system to control a process called spiral growth manufacturing (SGM), capable of building ten layers per minute. Despite these developments, most commercial LM machines are still slow for relatively large and complex prototypes.

1.2 Multi-Material Layered Manufacturing

Another major problem is that most LM machines to date can only fabricate homogeneous prototypes of a single material. However, recent trends in various industries, particularly advanced product development [9] and biomedical engineering [10], have warranted heterogeneous objects which offer superior properties unparalleled by homogeneous ones [1].   Heterogeneous objects may be classified into two major types, namely discrete multi-material (DMM) objects with a collection of distinct materials divided by clear boundaries, and functionally graded multi-material (FGM) objects with materials that change gradually from one type to another [11]. There is indeed an imminent need to develop multi-material layered manufacturing (MMLM) for fabrication of heterogeneous objects, and some pioneering works have been reported in recent years.

Qiu et al. [12] developed a virtual simulation system for fabrication of parts consisting of discrete materials; a toolpath planning method for two materials was reported to reduce defects and voids of a virtual part. Jepson [13] developed an experimental MMLM machine, which could blend two types of metallic powders to form a layer of some material gradients and subsequently sinter it to build an FGM part. Cho et al. [14] extended their patent "3D printing" to fabricate FGM parts; two materials were dispersed through their respective inkjet tools and printed into the powder bed.   Khalil et al. [10] developed a multi-nozzle biopolymer deposition system, which was capable of extruding biopolymer solutions and living cells for freeform construction of tissue scaffolds.   Cesarano III [15] developed a so-called Robocasting technology which was able to fabricate either single material or multi-material ceramic parts. By turning the blender on or off, fabrication of graded alumina / metal composites, and discrete placement of fugitive materials could be achieved.   Inamdar et al. [16] developed a multiple material stereolithography machine. The mechanism consisted of three vats, each of which contained a specific material, and a customised LabVIEW system was used to control the rotating multiple vat system to fabricate a multi-material model. Wang and Shaw [17] introduced a method for fabricating functionally graded materials

via inkjet colour printing. The print heads dispatched Al_2O_3 and ZrO_2 aqueous suspensions in different quantities to form a particular composition. Malone et al. [18] developed the Fab@Home multi-material 3D printer for fabrication of electro-mechanical systems. Typical materials included polypropylene and ABS thermoplastics, and low melting-point metal alloys such as lead and tin. A Zn-air cell battery of about the size of a coin, composed of five layers of different materials in a plastic case, was fabricated. Objet Geometries Ltd. [19] claimed that its Connex350 offered the ability to fabricate assemblies made of two types of photopolymer materials, with different mechanical properties. The photopolymer materials were cured by ultra-violet light immediately after jetting.

These systems have made significant contributions to the development of MMLM, although they were mostly experimental and could only make simple prototypes of a few types of materials. However, practical and viable MMLM systems for relatively large, complex objects have yet to be developed.

It can be said that development of MMLM is mainly concerned with three major research issues, namely (1) fabrication materials, (2) hardware mechanism for deposition of materials, and (3) computer software for planning the toolpaths and subsequent process control of multiple tools for prototype fabrication. These three issues are generally studied by researchers of specialised expertise. Nevertheless, the software issue of toolpath planning is particularly important as it has a significant impact on the overall efficiency and quality of fabrication, especially of large and complex prototypes.

1.3 Issues of Toolpath Planning

Toolpath planning for LM is mainly concerned with (i) contour filling strategy, and (ii) tool sequencing strategy [20]. Contour filling strategy concerns mainly with how to fill up the internal area of a contour. This issue has been well-studied and standard contour-filling patterns have been developed for LM [21]. On the other hand, tool sequencing strategy is more about coordinating the motions of a set of tools, each of which deposits a material on specific contours, to fabricate a multi-material prototype safely and effectively. Tool collisions and fabrication efficiency are main considerations, which may be exacerbated by the need to vary the tool deposition

speeds to suit different material properties and fabrication requirements [22]. Tools can be planned to deposit materials either sequentially to avoid collisions at the expense of fabrication efficiency, or concurrently to enhance fabrication efficiency with risks of collisions. This is a difficult problem of MMLM.

Few research works on toolpath planning for MMLM have been reported. Qiu et al. [12] developed a simulation system for toolpath analysis of MMLM. In the system, a toolpath file per material was generated first and then integrated into one multi-material toolpath file. A toolpath planning method for two materials was reported to reduce the defects and voids of a virtual part. This method could process relatively simple objects, such as cylinder and cube. Zhu and Yu [23] proposed a collision detection and tool sequencing method for simple multi-material assemblies. Zhou [24] proposed a toolpath planning algorithm for fabrication of functionally graded multi-material (FGM) objects. First, the gradual material distribution in each layer was discretised into step-wise sub-regions, in each of which the material could be assumed homogeneous. Then, sequential toolpath for each sub-region was generated separately.

The experimental MMLM systems described in the previous section also involved some basic toolpath planning algorithms which were either sequential or could only handle relatively simple prototypes. Choi and Cheung [25] developed a multi-material virtual prototyping system integrated with a topological hierarchy-based approach to toolpath planning for MMLM. This approach was later improved with an entire envelope-based approach [20] and with a separate envelope-based approach [26]. These approaches were characterised by the construction of bounding envelopes around slice contours by offsetting outward a distance of the tool radius. Overlap test was executed for these envelopes. Tools in the non-overlapped envelopes could deposit their specific materials concurrently without any collisions. Nevertheless, they did not allow tools to move concurrently when the associated envelopes of the contours overlapped, incurring some idle time of tools.

1.4 Research Objective

It can be concluded that toolpath planning for MMLM remains a vital but difficult research issue, which has yet to be fully tackled. This paper therefore proposes a new approach to concurrent toolpath planning for MMLM to further improve the

fabrication efficiency of relatively large, complex prototypes. This approach eradicates the associated constraints of the previous approaches [20, 25, 26] to further improve the fabrication efficiency. It is characterised by construction of envelopes around individual tools directly, rather than around the slice contours of each layer. Relative tool positions are monitored to detect potential collisions at a predefined rate. A dynamic priority assignment scheme is applied to assign motion priorities of the tools to avoid collisions and to coordinate the tool motions accordingly. Deposition speeds of tools can also be adjusted to suit different material properties and fabrication requirements. This concurrent toolpath planning approach can substantially shorten the build-time of MMLM, in comparison with the previous approaches.

2 Related Works

2.1 Collision Detection

Tool collision is a major obstacle of multi-toolpath planning. Collision detection, also known as interference detection or contact determination, is an interdisciplinary issue which is particularly important in motion planning, robotics, CAD/CAM, etc [27]. Detection accuracy and computation cost are two inherently contradictory factors. The rate of collision detection is quite application-specific. For instance, haptic interfaces require update rates of about one thousand hertz, while about twenty to thirty updates per second would be sufficient for real-time graphical applications [28]. A number of application-specific and practical collision detection algorithms have been proposed, and each of them has its own merits and deficiencies [29].

2.2 Multi-Object Motion Planning

In an MMLM process, a number of tools deposit materials on specific contours to fabricate a prototype, preferably in concurrent motion to increase efficiency. It can be regarded as a more general control problem of multiple mobile objects sharing a common workspace to complete their individual tasks without collision. This problem has received a great deal of attention in other applications, such as mobile robots [30], manipulation of robot arms [31], route planning for vehicles in a warehouse [32], etc.

A variety of methods have been proposed to solve the multi-object motion planning problem, in which avoidance of collisions is paramount for safety and effectiveness. In

general, potential collisions are first detected, and the object motions are subsequently coordinated to avoid collisions and to improve optimality. Optimisation objectives include the minimisation of energy, path length, and motion time. Indeed, practicality often demands a good balance between optimality (solution quality) and complexity (computation cost). A broad review of multi-object motion planning can be found in Lavalle's work [33], and this problem may be roughly categorised into coupled and decoupled [34], as follows.

2.2.1 Coupled Methods

A coupled method for multi-object motion planning combines the configuration spaces of all the objects into a composite configuration space in which a feasible path is searched for [32]. In general, it can achieve completeness and optimality. However, the composite configuration space grows exponentially with the number of objects [35], rendering the problem PSPACE-hard [36]. Lavalle and Hutchinson [37] worked on simultaneous optimisation of the motions of three robots from the start points to the goals. Li and Latombe [31] presented an approach to concurrent manipulation of two robot arms to grab parts of various types on a conveyor and transfer them to their respective goals while avoiding collisions with obstacles. These works were among early applications of coupled multi-object motion planning techniques on relatively simple systems. Indeed, coupled methods are often used in systems with only a few objects, or for off-line applications.

2.2.2 Decoupled Methods

In a decoupled method, on the other hand, the path of each object is separately generated and subsequently coordinated to avoid collision [32]. Different decoupled methods have been applied to coordinate the motion of multiple objects, including adjustment of geometric paths, modification of velocities, and time delay [34]. Wagner et al. [30] demonstrated an efficient approach to coordinating a group of cooperative cleaner robots to clean a common dirty floor. Lee and Kim [38] developed a multi-robot printing system, in which the host computer commanded a set of client printer-robots to cooperatively draw a picture on a sheet of paper. In the work of Peng and Akella [39], the path of each robot was first generated irrespective of other robots, while the velocities were subsequently altered along their paths to avoid collision. Rekleitis et al. [40] presented an algorithm to control a team of robots, moving in

zigzags, to complete the coverage of a 2D plane. Lee et al. [41] used potential functions to guide objects to their destinations. Chang et al. [42] used a time delay method to avoid collisions between two robot arms. The minimum delay time needed for collision avoidance was obtained by a collision map scheme.

In essence, decoupled methods involve assigning priorities to objects to determine the order in which the paths are to be coordinated [43]. Erdmann and Perez [44] assigned static priority to each robot and sequentially computed paths in a time-varying configuration space. Ferrari et al. [45] used a fixed priority scheme and chose random detours for the robots with lower priorities. These static priority schemes were suitable for predefined applications. On the other hand, dynamic priority schemes are more flexible to handle different situations. Azarm and Schmidt [46] proposed an approach that considered all possible priority assignments for up to three robots. Clark et al. [47] presented a motion planning system that could construct collision-free paths for groups of robots in dynamic environments. They introduced a priority scheme that gave way to the robot whose local workspace was most crowded. Bennewitz et al. [35] optimised different possible priority schemes for teams of mobile robots. Unfortunately, searching different sequences of priorities was computation-intensive, and might fail to find solutions to complex planning problems. van den Berg and Overmars [43] proposed a heuristic for assigning priorities to a team of robots, in which a higher priority was assigned to a robot with a longer moving distance. Decoupled methods often adopt priority-based approaches due to its prioritisation essence [35]. They are computationally simpler and can response faster in real-time applications, in comparison with coupled methods, although they cannot guarantee optimal solutions. Moreover, they are scalable for handling more mobile objects [32].

Based on the review above, a dynamic priority-based decoupled method is proposed to generate concurrent toolpaths for MMLM. However, MMLM has its distinct characteristics with respect to multi-object motion planning. First, the tools do not have self-control and sensing capabilities, and they cannot communicate with each other. They are coordinated by a central controller. Second, the tools are normally constrained to move along fixed paths of zigzags or spirals, at constant deposition speeds. And third, the toolpath planning approach for MMLM should take into account the mechanical and thermal properties of the fabrication materials. Hence, the

multi-object motion planning technique has to be suitably adapted for concurrent toolpath planning for MMLM.

3 The Dynamic Priority-based Approach to Concurrent Toolpath Planning for MMLM

In a general scenario of multi-objects, the object motions may be omni-directional and erratic with multiple intersections, as shown in Figure 1. But tools in MMLM are normally constrained to move along fixed paths of zigzags or spirals at constant deposition speeds inside specific contours that do not overlap, as shown in Figure 2. The material deposition mechanism of MMLM may consist of a set of tools, each of which deposits a specific material on the related slice contours. The tools do not invade other unrelated contours, and there may be collisions of tools only when they get in close vicinity, like the purple tool and the yellow tool. Indeed, some hardware constraints may hinder concurrent deposition of materials and consideration of collisions between tools and support mechanisms, in addition to tool collisions, would further complicate the control problem. But as pointed out previously, hardware mechanism is another research issue of MMLM. We therefore assume appropriate deposition hardware without tool-support interferences would be available, and we limit the scope of this paper to consider collisions between tools only.
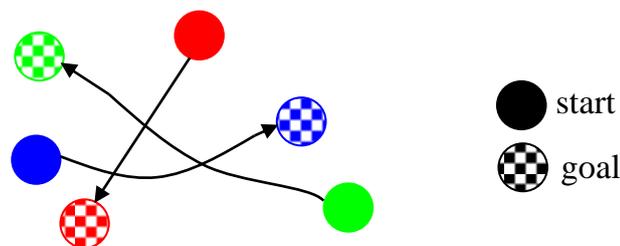

Figure 1 A general motion scenario of multiple objects

Model of a human eye ball in skull

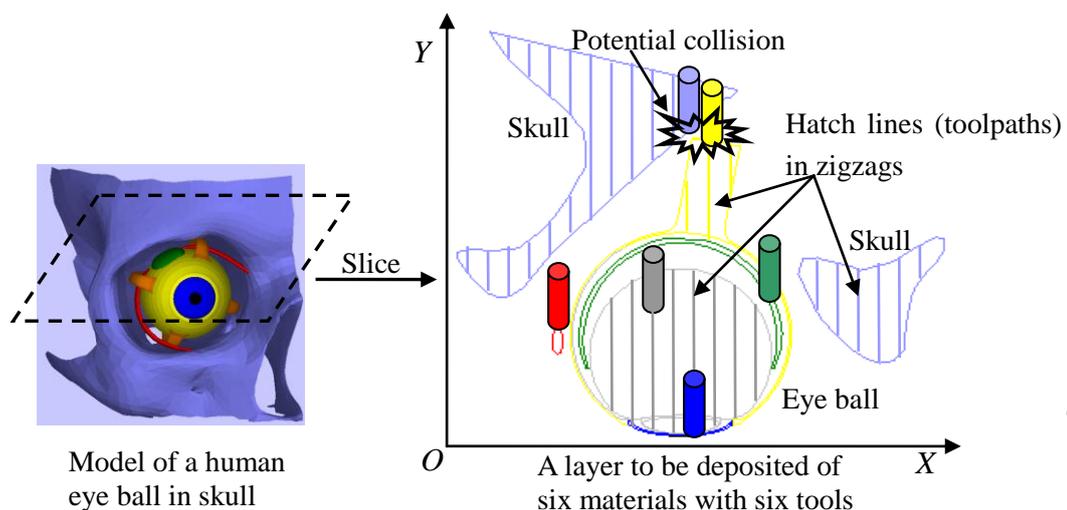A layer to be deposited of six materials with six tools

9

Figure 2 Schematic of the proposed MMLM mechanism

The proposed concurrent toolpath planning approach first generates the toolpath (hatch lines) of each tool for depositing a specific material on the related contours of a layer. The tools are then coordinated to fabricate the layer concurrently. Collision detection of tools is executed at a predefined rate. When a potential collision between a pair of tools is detected, their traverse speeds along the x-axis are compared, and a higher priority is assigned to the tool travelling at a higher traverse speed. The tool with a higher priority continues to deposit material along its original path, while the one with a lower priority gives way by pausing at a suitable point until the potential collision is eliminated. As such, the level of concurrent tool motions, and hence the overall fabrication efficiency, can be significantly improved. Moreover, the deposition speeds of tools can be adjusted to suit different material properties and fabrication requirements. The following sections present the details of traverse speed, collision detection, and motion priority assignment for implementation of the proposed concurrent toolpath planning approach.
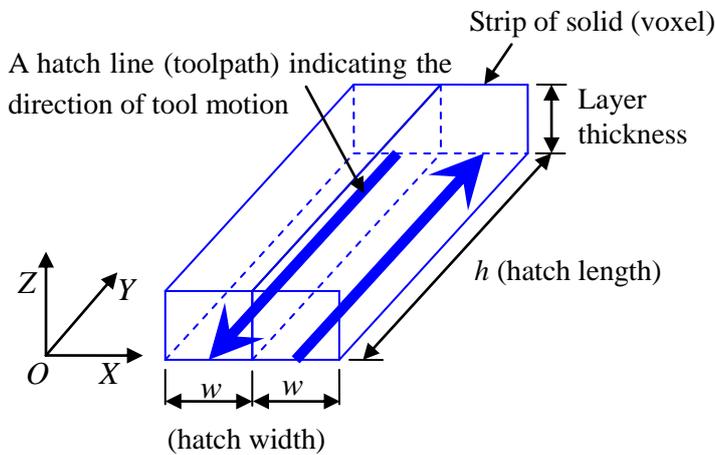
3.1 Analysis of Traverse Speed: V_x

There are two common modes in which a tool fills up a contour: (1) the zigzag mode where the toolpaths are hatch lines which can be either horizontal, or at 45° slope, or vertical [12] and, (2) the spiral mode where the toolpaths are offset inwards from the contours. The proposed approach adopts the zigzag mode with vertical hatch lines. As shown in Figure 3(a), a tool moves in up-and-down zigzags at deposition speed V to deposit material to fill up a circular contour. As a whole, the tool moves from the left to the right at a traverse speed V_x. The deposition speed V is bidirectional, while the traverse speed V_x is unidirectional. V can be varied to suit different material properties and fabrication requirements, but it is constant for a specific material during the whole fabrication process of a prototype. V_x, on the other hand, depends on the hatch lines. Figure 3(b) shows the lengths and the widths of two adjacent hatch lines. While the width w remains constant, the length h varies across the contours during the fabrication process. Assume that a tool traverses a distance of the hatch width w along the x-axis in a time it takes to complete depositing material along the hatch length h. The time that a tool spends on a hatch line is $T = h/V$, and therefore the tool's traverse speed is given by $V_x = w/T = wV/h$. It can be seen in Figure 3(a) that the hatch length increases from the leftmost to the centre of the circle and decreases from the centre to the rightmost of
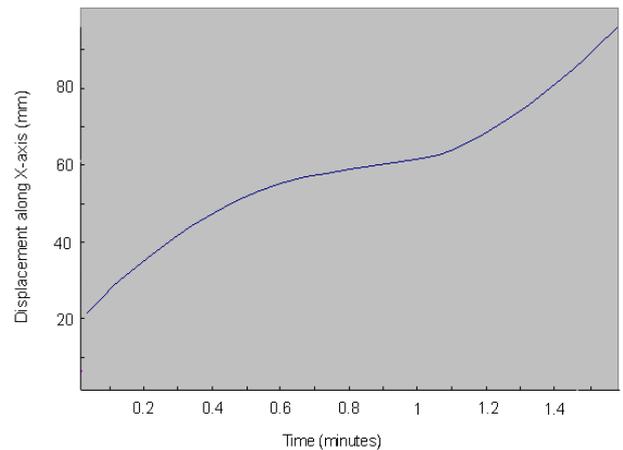
the circle afterwards. Accordingly, the traverse speed V_x of a tool decreases first and increases afterwards. The curve of tool displacement along X-axis versus time is shown in the *X-t* graph in Figure 3(c), whose varying gradient is V_x.
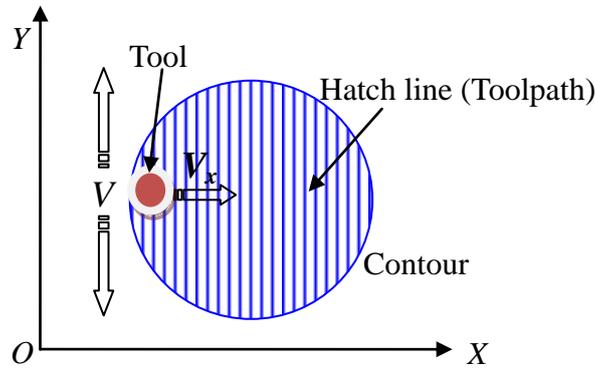
(a) A simple layer of a circular contour

(b) Hatch length and hatch width

(c) *X-t* graph of the circular contour

Figure 3 Analysis of traverse speed

It can be said that the traverse speed V_x of a tool is instantaneous, varying across a contour during the fabrication process. Motion priorities are dynamically assigned to the tools according to their traverse speeds to avoid collisions.

3.2 Collision Detection

Collision detection plays an important role in avoiding collisions to ensure the safety and effectiveness of an MMLM process. Since the tools deposit specific materials at related slice contours concurrently, there may be collisions between a pair of tools when they get in close vicinity. In multi-object planning, object motions may be omni-directional and erratic with multiple intersections, rendering the relative velocities and the motion directions of objects complex factors in collision detection. It

11

may be necessary to iterate detection of the instantaneous relative velocities and centre distances of tools, which is computationally intensive if the detection rate is high.

However, as we adopt up-and-down zigzags as the contour filling strategy, detection of tool collisions for MMLM can be simplified considerably. It would only be necessary to consider the distance between the ends of the hatch lines being deposited by the tools in question. In general, collisions between a pair of tools would not happen if either the horizontal or the vertical distance between the ends of two hatch lines is greater than the sum of the tool radii, regardless of the deposition speeds and directions of the tools. Hence, collision detection is only needed at a predefined rate of the completion of a number of hatch lines, making the algorithm relatively simple yet effective. The principle of the proposed collision detection is outlined as follows.

A cylinder is constructed around each tool as the bounding envelope and hence a circle projected on the X-Y plane represents a tool. In Figure 4(a), a red tool and a blue tool are depositing materials along hatch lines AB and CD, respectively. The tools are of the same radius R. Let dx be the horizontal distance between the ends of two hatch lines which are currently being deposited by the associated tools respectively, and dy the closest vertical distance between the ends of the two hatch lines. Since $dx < 2R$ and $dy < 2R$, the tools are about to collide when the red tool is near point B and the blue tool is close to C.

Condition of potential collision:
Any two tools moving along their respective hatch lines are considered as about to collide if the following condition holds:

$$dx \leqslant 2R + safety\ margin = 2R + R = 3R \quad \text{AND} \quad dy \leqslant 2R.$$

The proposed algorithm incorporates a safety margin of R in the x-axis direction, which may be changed if necessary, for further safeguard against potential collisions as shown in Figure 4(b). Figures 4(c) and 4(d) show two cases in which the tools would not collide when $dx > 3R$, and when $dx < 3R$ but $dy > 2R$, respectively.

The rate of collision detection should be well chosen to strike a good balance between detection accuracy and computation cost. A high detection rate improves accuracy with

more computation resources, while a low detection rate reduces computation cost at the expense of accuracy. For the MMLM process, the highest rate is to detect collision after fabrication of a hatch line. On the other hand, we adopt the lowest allowable rate of collision detection, which is derived as follows. In Figure 4(b), imagine an extreme situation in which the traverse speed V_{xB} of the blue tool approaches zero, the red tool will catch up with the blue tool after the completion of $n = R/w$ hatch lines, where R is the safety margin and w is the hatch width. Hence, an interval of n hatch lines is the lowest allowable rate of collision detection. Whenever a tool first completes n hatch lines, the collision detection is executed.
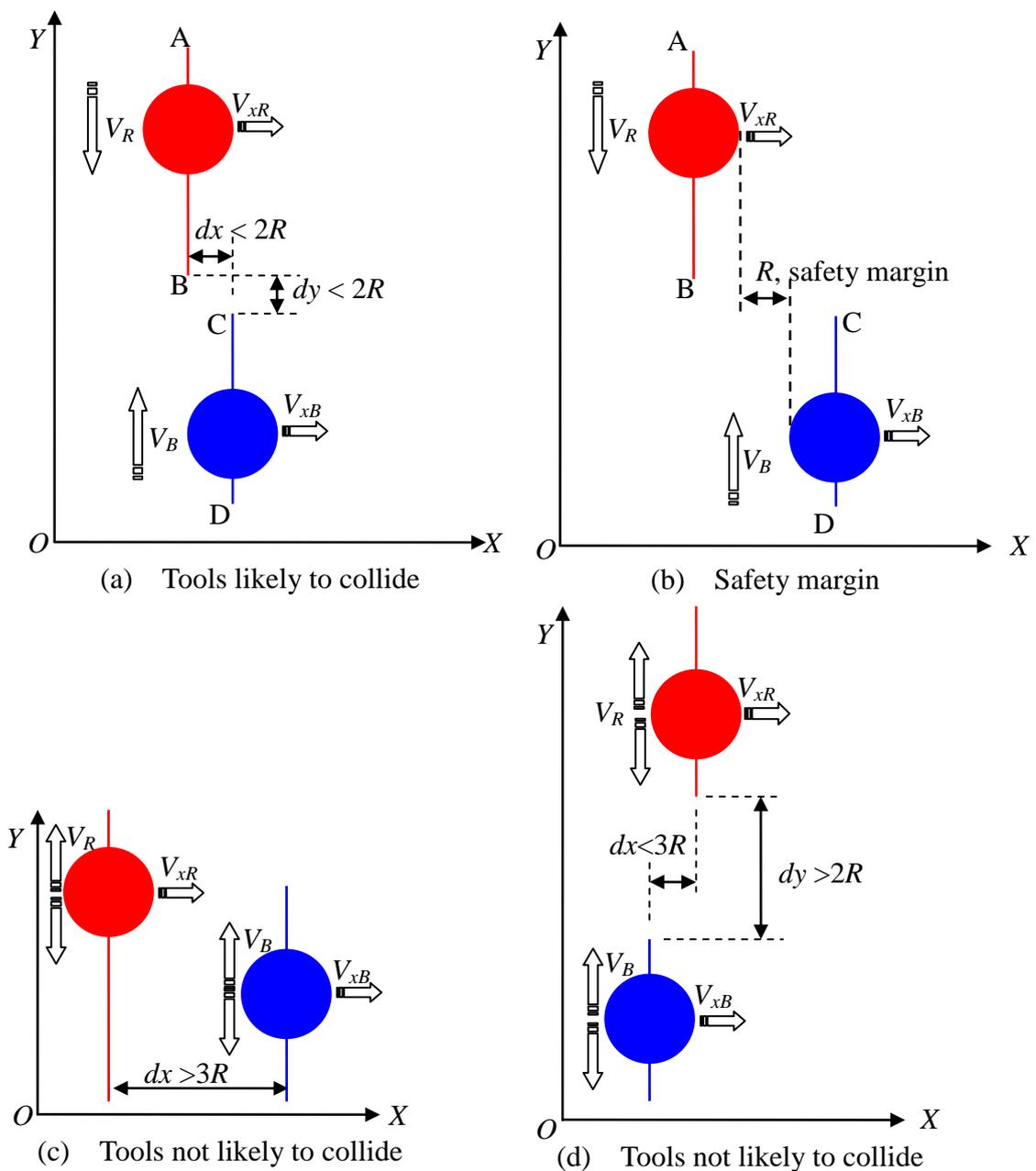

(a) Tools likely to collide

(b) Safety margin

(c) Tools not likely to collide

(d) Tools not likely to collide

Figure 4 Tool collision detection for MMLM 13

3.3 Dynamic Assignment of Priorities for Tool Motion Coordination

When potential collision between a pair of tools is detected, a higher motion priority is assigned to the tool that travels at a higher traverse speed for it to continue to deposit material along its original path. The tool with a lower priority gives way by waiting at a suitable point until the potential collision is eliminated. This priority scheme ensures a potential collision is resolved as quickly as possible so that the paused tool can resume fabrication to minimise uneven cooling that may impair the prototype quality. The procedure of dynamic assignment of tool motion priorities is as follows:

Step 1: Read in a new layer, initialise the speeds of tools, and start fabrication;
Step 2: Perform collision detection between all pairs of the tools;
Step 3: If no potential collision is detected, go to Step 8;
Step 4: Find the pair of tools which is likely to collide;
Step 5: Calculate the traverse speeds $V_x = wV/h$ of tools which are likely to collide;
Step 6: Assign priorities to the tools according to their traverse speeds. A higher priority is assigned to a tool at a higher V_x;
Step 7: A tool with a higher priority continues deposition along its original path, while the one with a lower priority waits at a suitable point to give way until the potential collision is eliminated;
Step 8: Tools without potential collision deposit materials concurrently;
Step 9: If a layer is not completed, repeat from Step 2. Otherwise, repeat from Step 1 until all the layers are completed.

3.4 A Simple Prototype to Illustrate the Proposed Approach

The following section illustrates how the proposed dynamic priority-based approach to concurrent toolpath planning for MMLM works.

3.4.1 Workflow of the Proposed Concurrent Toolpath Planning Approach

Figure 5 shows a simple prototype, of dimensions 230mm x 100m x 6mm, to be made of three discrete materials. The prototype is sliced into 30 layers. In Figure 6, three tools, T_{red}, T_{green}, and T_{blue}, of the same radius $R=10$mm, deposit the red, green, and blue materials respectively on a selected layer. For simplicity, the deposition speeds of the three tools are set to be $V_R = V_B = V_G = 10$mms^{-1}, and the hatch width w is 1mm. The effect of adjusting the deposition speeds will be presented in the next section.
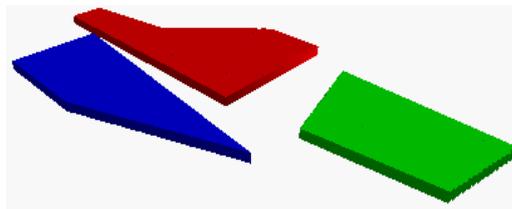

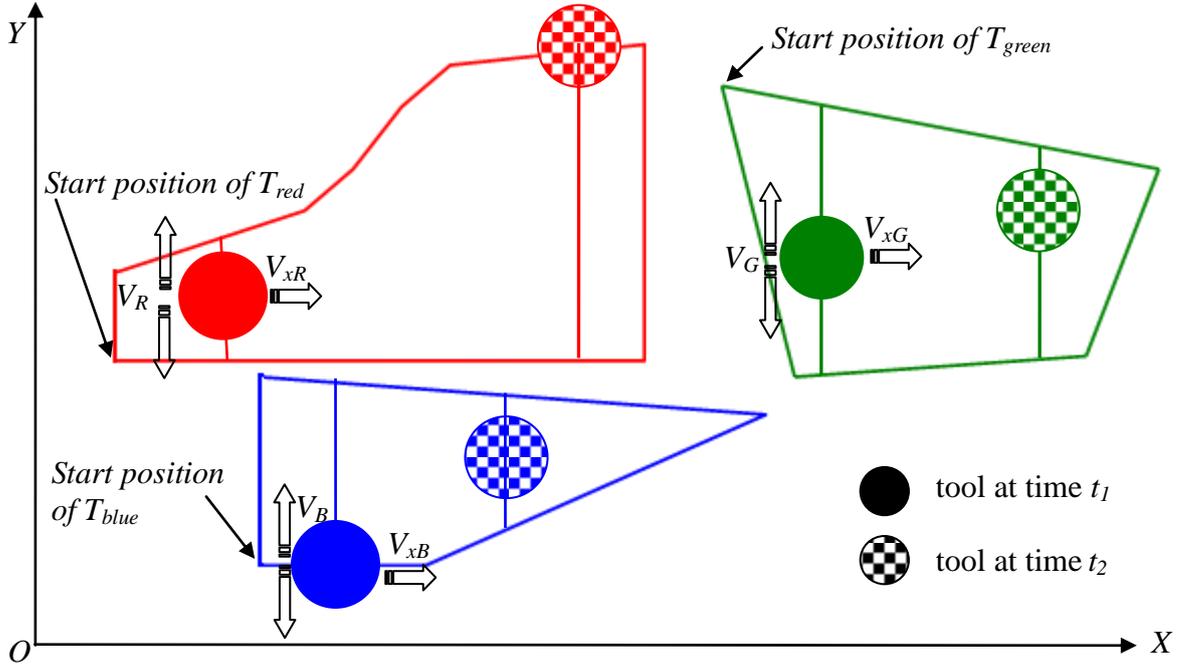Figure 5 A simple prototype of three discrete materials

Figure 6 Three tools fabricating the selected layer

To start fabricating this layer, the three tools are initialised with their deposition speeds and they move from their datum positions around O to the bottom point of the leftmost hatch line of their respective contours. The proposed collision detection algorithm is executed on all pairs of the tools. At first, there is no potential collision as dx between all pairs of current hatch lines are greater than $3R$, so all the three tools can start to deposit specific materials concurrently. The traverse speed of each tool is $V_{xR} = V_{xG} = V_{xB} = wV/h$, where w and h are the width and length of the current hatch line being deposited by each tool.

As shown in Figure 6, the length of each red hatch line is at first shorter than that of the blue one; the traverse speed of the red tool is thus higher than that of the blue one, i.e., $V_{xR} > V_{xB}$.   The red tool will catch up with the blue one some time later. As shown in Figure 7, collision detection indicates that there is a potential collision between the red tool and the blue tool at time t_1, because the condition of $dx < 3R$ and $dy < 2R$ holds. The dynamic priority assignment algorithm is executed here to adjust the motion priorities of the red tool and the blue tool accordingly.   It calculates and compares the traverse speeds of the tools at their respective current hatch lines, and assigns priorities to these tools according to their traverse speeds. Here, a higher priority is assigned to the red tool, for it is at a higher traverse speed. The red tool continues its original path, while

15

the blue tool with a lower priority waits at a suitable point to give way until the potential collision is eliminated. As to the blue tool, waiting at a suitable point means that the blue tool waits at the far end of its current hatch line after completing deposition (shown as the solid blue circle in Figure 6) in order not to hamper the quality of the resulting prototype.
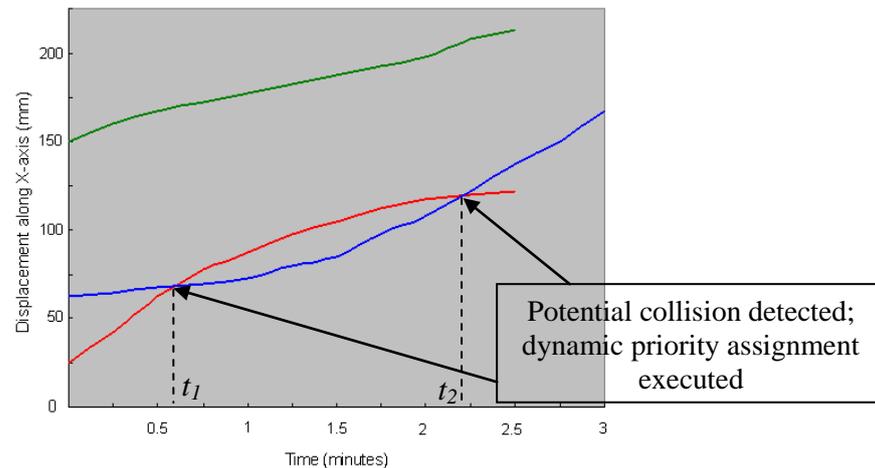

Figure 7 *X-t* graph of the selected layer

Now, while the blue tool is paused, the red tool and the green tool deposit materials concurrently, because no potential collision between them is detected. Some time after t_1, the collision detection algorithm finds that the potential collision between the red tool and the blue tool has been eliminated. Subsequently, all the three tools are commanded to fabricate concurrently.

Some time after the tools gradually traverse to the right side of the respective contours, the length of the blue hatch line becomes shorter than that of the red one. Thus, the traverse speed of the blue tool becomes higher than that of the red tool and the blue tool is catching up with the red tool. At time t_2, as shown in dashed circles in Figure 6 and time t_2 in Figure 7, collision detection finds that there is a potential collision between the red tool and the blue tool again.

Now, the traverse speed of the blue tool overtakes that of the red tool. The priorities of these two tools are reversed, i.e., a higher priority is assigned to the blue tool. Thus, the blue tool continues its original path while the red tool with a lower priority waits at a suitable point to give way until the potential collision is eliminated.   At this moment,

the blue tool and the green tool, without potential collision, deposit concurrently while the red tool is waiting. Some time after t_2, the collision detection algorithm finds that the potential collision between the red tool and the blue tool has been eliminated. All the three tools can again fabricate concurrently until the layer is completed. Digital fabrication of the selected layer with $V_R = V_B = V_G = 10\text{mms}^{-1}$ is shown from Figure 8(a) to Figure 8(e).
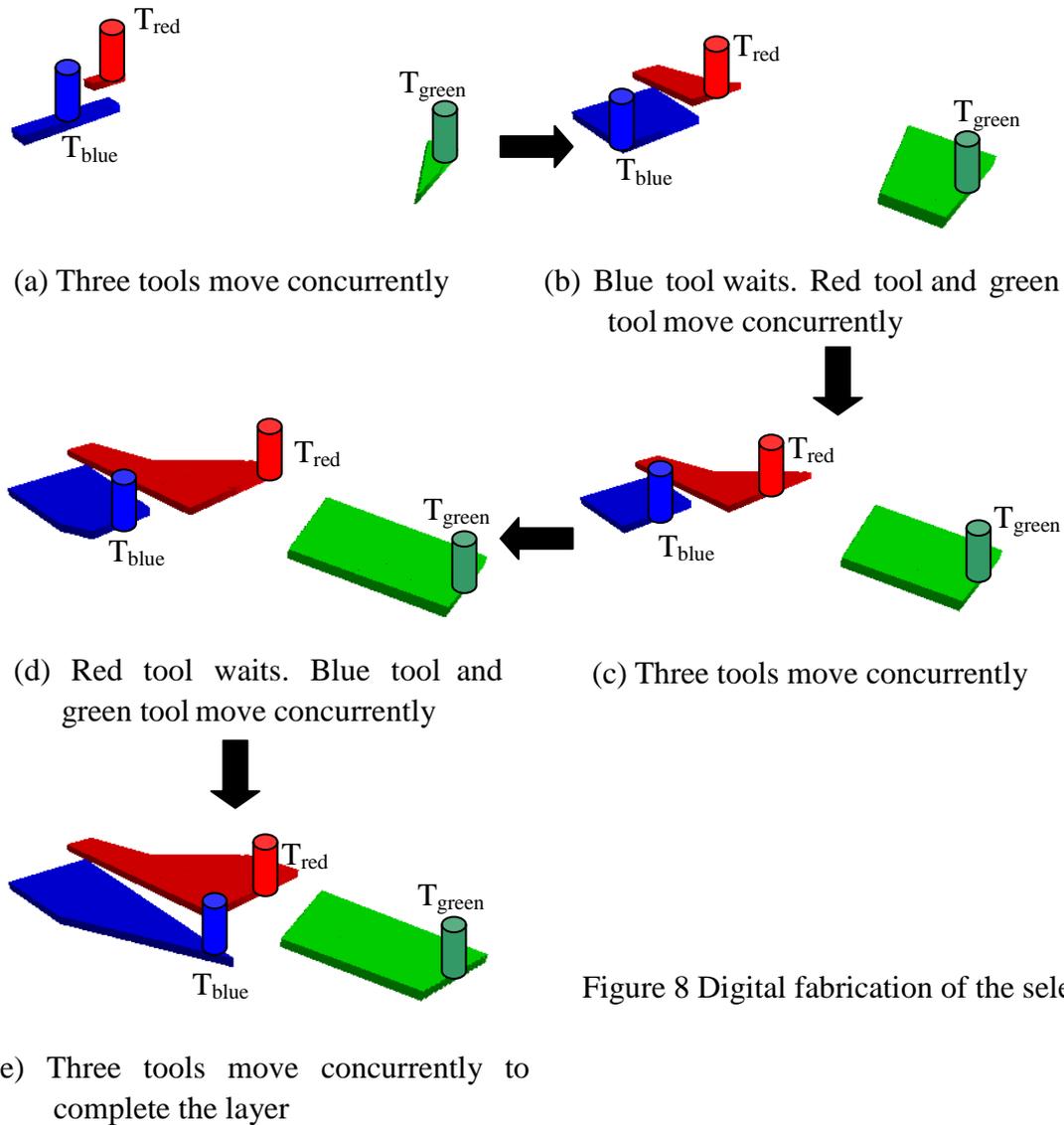
(a) Three tools move concurrently

(b) Blue tool waits. Red tool and green tool move concurrently

(d) Red tool waits. Blue tool and green tool move concurrently

(c) Three tools move concurrently

Figure 8 Digital fabrication of the selected layer

(e) Three tools move concurrently to complete the layer

3.4.2 Adjustment of Deposition Speeds

To suit different material properties and fabrication requirements, it may be necessary to adjust the deposition speed of a tool. In this section, the deposition speeds of the red, green, and blue tools are varied to be $V_R = 15\text{mms}^{-1}$, $V_G = 10\text{mms}^{-1}$, $V_B = 8\text{mms}^{-1}$,

17

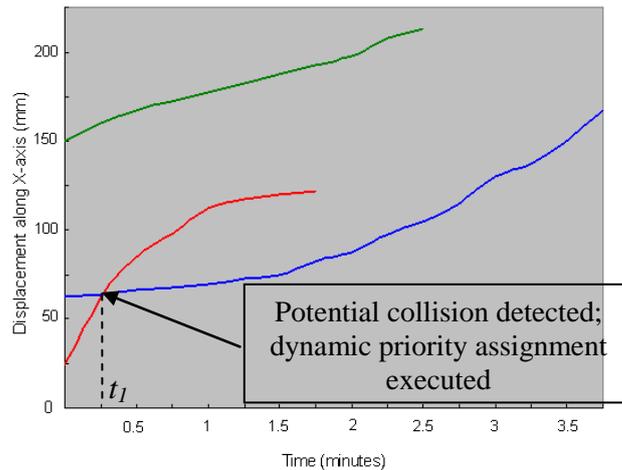respectively.   Figure 9 shows the X-t graph with these adjusted deposition speeds.

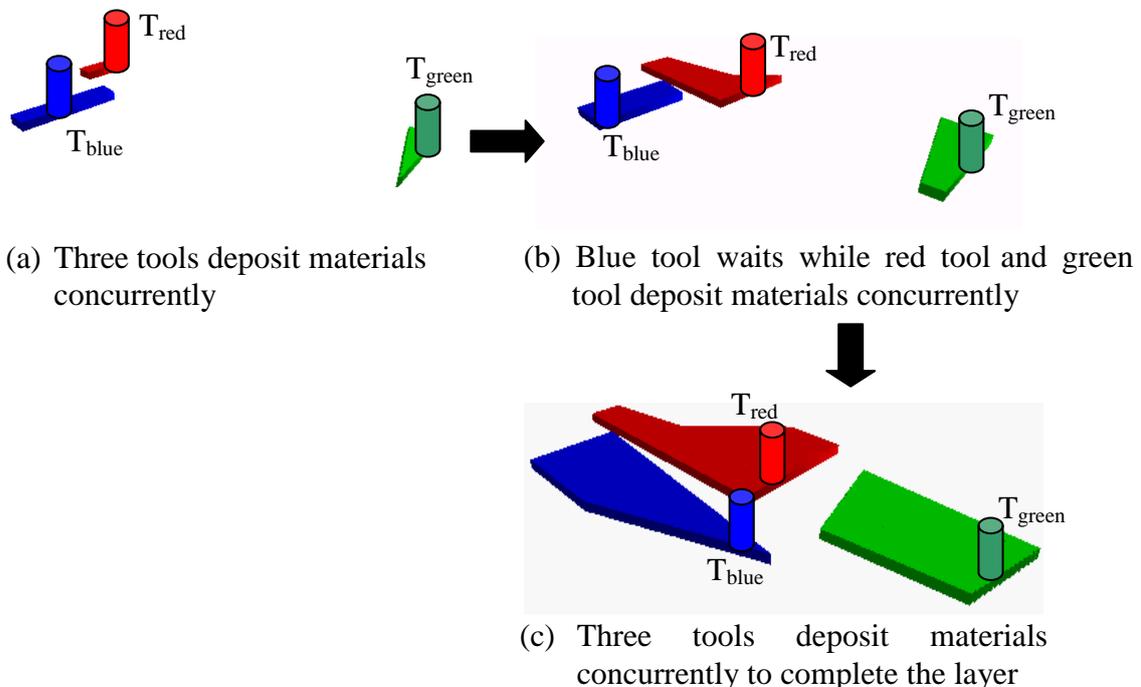Figure 9 *X-t* graph of the selected layer with adjusted deposition speeds

(a) Three tools deposit materials concurrently

(b) Blue tool waits while red tool and green tool deposit materials concurrently

(c) Three tools deposit materials concurrently to complete the layer

Figure 10 Digital fabrication of the selected layer at adjusted deposition speeds

Like in the previous example, the three tools first move concurrently to deposit their specific materials as shown in Figure 10(a). Then, due to the higher deposition speed of the red tool and the shorter length of red hatch lines, the traverse speed of the red tool is much higher than that of the blue one. Hence, the red tool will quickly catch up with the blue one at time t_1, as shown the *X-t* graph in Figure 9. The dynamic priority assignment algorithm is executed for the red tool and the blue tool accordingly. A higher priority is assigned to the red tool which is at a higher traverse speed. The red

18

tool continues fabrication along its original path, while the blue tool with a lower priority waits at a suitable point to give way until the potential collision is eliminated, as shown in Figure 10(b).

Now, the red tool and the green tool, which are not likely to collide, deposit materials concurrently while the blue tool is waiting. Some time after t_1, the collision detection algorithm finds that the potential collision between the red tool and the blue tool has been eliminated. The three tools can again deposit materials concurrently, as shown in Figure 10(c).   Hence, in comparison with the previous example of uniform deposition speed, the red tool travels a lot faster and it will never be caught up with by the blue tool after t_1. All the three tools continue to deposit materials concurrently until the layer is completed.

4 Implementation and Case Studies

The proposed dynamic priority-based approach has been incorporated with other major in-house modules for STL model manipulation, slicing, hierarchical contour sorting, contour hatching, and digital fabrication, to form an integrated system for multi-material virtual prototyping (MMVP) [48].   The system was implemented in C/C++, and integrated with WorldToolKit Release 9 for fabrication simulation in a semi-immersive virtual reality (VR) environment, and with Virtools Dev toolkits for a full-immersive CAVE VR environment.   It can digitally fabricate relatively complex objects for biomedical applications and advanced product development. The following presents two case studies, a human ear model and a toy tank, to demonstrate some possible applications of the MMVP system and to verify the effectiveness of the proposed dynamic priority-based approach to concurrent toolpath planning for MMLM.

4.1 A Human Ear Model

In recent years, doctors and surgeons have often used biomedical prototypes to help visualise the anatomy of human organs and design prostheses for surgical planning and implantations. Indeed, multi-material prototypes would be particularly useful for study and planning of delicate surgeries, in that they can differentiate clearly one part from another, or tissues from blood vessels of a human organ.   A model of a human ear with dimensions of 175mm x 186mm x 190mm, as shown in Figure 11, is sliced into

800 layers with layer thickness of 0.2mm. Figure 12 shows the contours of a layer to be made of four materials coloured in orange, pink, blue, and grey, with four tools of the same radius $R=10$mm.

The deposition speeds are set to be $V_P =20$mms^{-1}, $V_O =15$mms^{-1}, $V_B =5$mms^{-1}, and $V_G=5$mms^{-1} for the tools that deposit pink, orange, blue, and grey materials, respectively. Adjusted deposition speeds will be presented later.
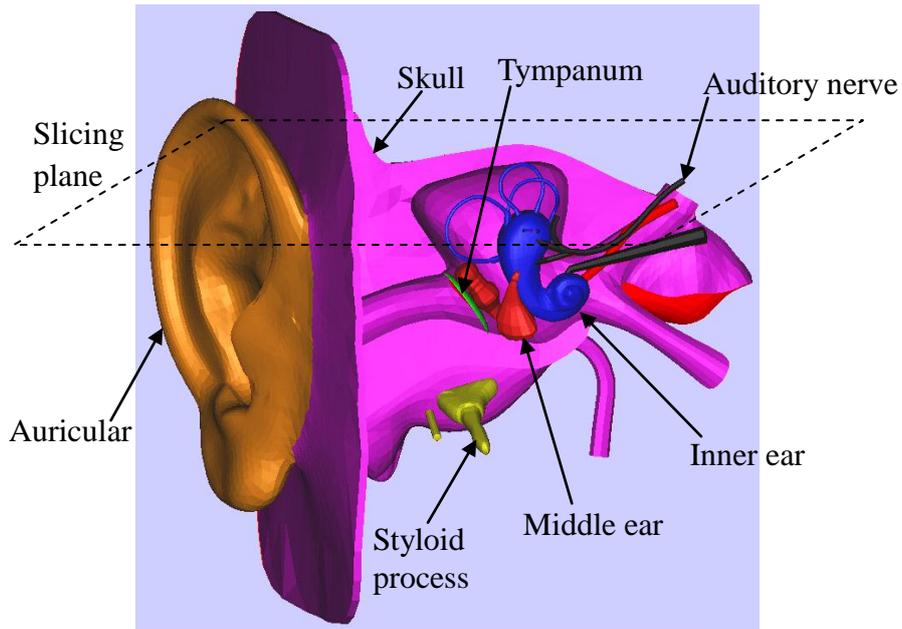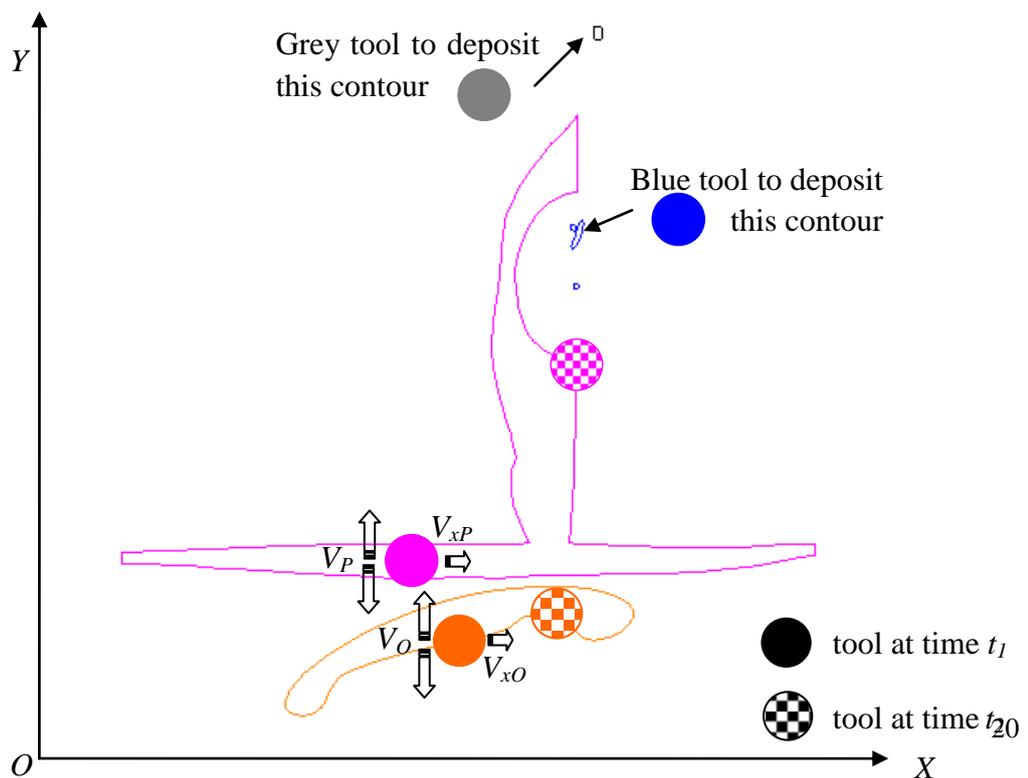

Figure 11 Anatomical model of a human ear

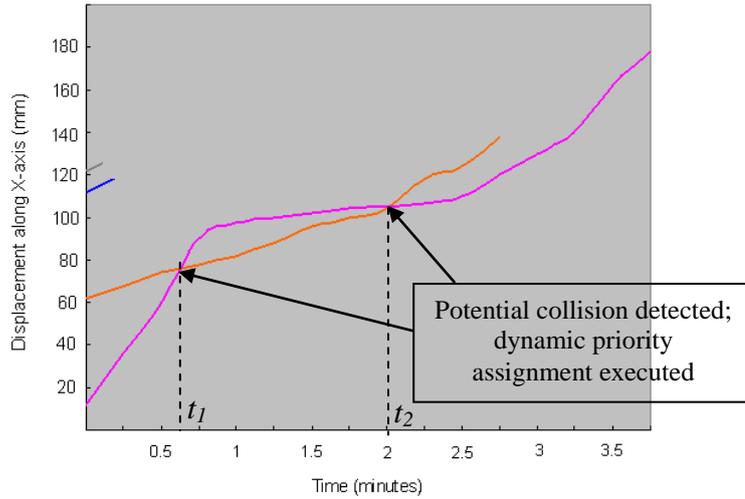Figure 12 A selected layer of the human ear model

Figure 13 *X-t* graph of the selected layer of the human ear model

At first, the pink, orange, blue, and grey tools move concurrently to deposit their specific materials. When $t = t_1$, as shown in solid circles in Figure 12 and the time t_1 in Figure 13, collision detection finds that there is a potential collision between the pink tool and the orange tool. The dynamic priority assignment is executed for the pink tool and the orange tool according to their traverse speeds. A higher priority is therefore assigned to the pink tool which is at a higher traverse speed. This allows the pink tool to continue deposition along its original path, while the orange tool with a lower priority waits at the far end of its current hatch line until the potential collision is eliminated. As collision detection is executed at the predefined rate, the potential collision between the pink tool and the orange tool has been eliminated some time after t_1. Hence, they can again deposit materials concurrently, as shown in Figure 13 some time after t_1. Meanwhile, it can be noticed that the blue tool and the grey tool have already completed their respective tasks.

At time t_2, as shown in dashed circles in Figure 12 and time t_2 in Figure 13, a potential collision is detected between the pink tool and the orange tool again. Contrary to the case at t_1, the pink tool now traverses at a lower speed than the orange tool, because the lengths of hatch lines of the pink contour are much longer than that of the orange one. The dynamic priority assignment is executed again for the pink tool and the orange tool in the order of their traverse speeds. Thus a higher priority is assigned to the orange tool which is at a higher traverse speed. The orange tool continues to deposit material along its original path, while the pink tool with a lower priority waits until the

potential collision is eliminated.　　Some time after t_2, the potential collision between the pink tool and the orange tool is found to have been eliminated. This pair can again deposit materials concurrently, as shown in Figure 13 some time after t_2. When the pink tool finishes its task, this layer is completed.　　The digital fabrication process of the selected layer of the ear model is shown in Figure 14.
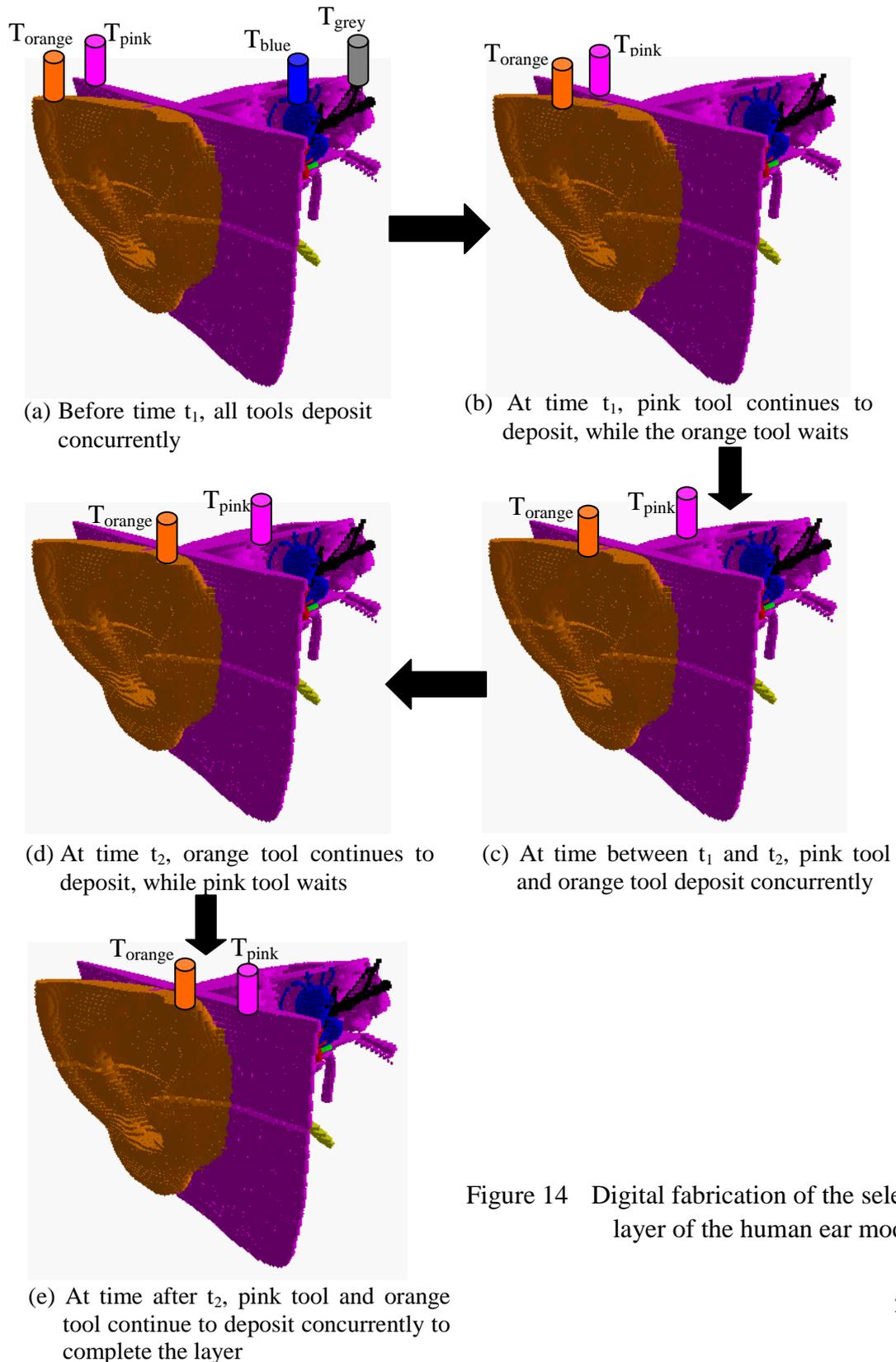

(a) Before time t_1, all tools deposit concurrently

(b) At time t_1, pink tool continues to deposit, while the orange tool waits

(d) At time t_2, orange tool continues to deposit, while pink tool waits

(c) At time between t_1 and t_2, pink tool and orange tool deposit concurrently

(e) At time after t_2, pink tool and orange tool continue to deposit concurrently to complete the layer

Figure 14　Digital fabrication of the selected layer of the human ear model
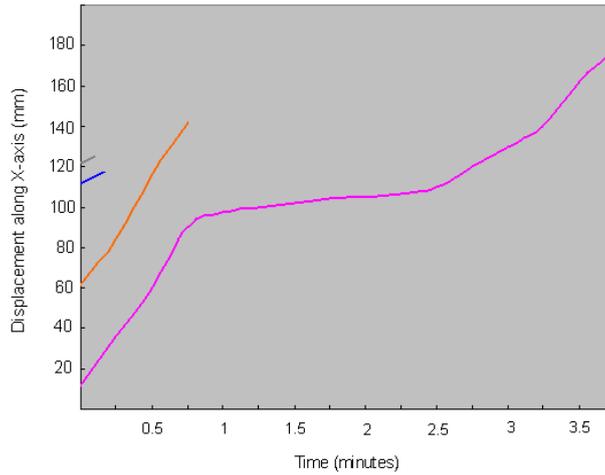
Figure 15 *X-t* graph of the ear model layer with increased deposition speed of the orange tool
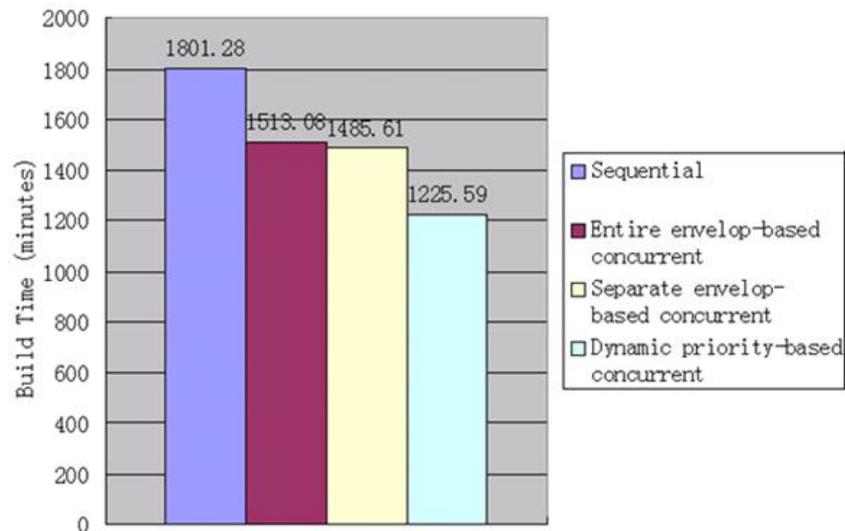

Figure 16 Comparison of build times of the human ear model by different toolpath planning approaches

To suit different material properties and fabrication requirements, it is assumed that the deposition speed of the orange tool is increased from 15mms^{-1} to 20mms^{-1}. Figure 15 shows *X-t* graph of the ear model layer with increased deposition speed for the orange tool. In this case, the pink tool can never catch up with the orange tool. Hence, four tools can deposit concurrently to complete the layer. Figure 16 compares the build times of the human ear model by different toolpath planning approaches. For consistency, the deposition speeds of tools in all the approaches are V_P =20mms^{-1}, V_O =15mms^{-1}, V_B =5mms^{-1}, and V_G=5mms^{-1}. It can be seen that the proposed dynamic priority-based approach improves the efficiency by 32%, 19%, and 18% respectively,

23

in comparison with the sequential approach, the entire envelope-based approach [20], and the separate envelope-based approach [26]. The entire envelope-based and the separate envelope-based approaches are characterised by the construction of bounding envelopes around slice contours by offsetting outward a distance of the tool radius. Overlap test is executed for these envelopes, and tools are not allowed to move concurrently when the associated contour envelopes overlapped. For example, the orange tool in Figure 12 cannot move until the pink tool completes the pink part.

The proposed dynamic priority-based approach eliminates this constraint. It constructs envelopes around the tools directly, rather than around the slice contours. Relative positions of tools are monitored at a predefined rate. It enhances concurrency of tool motions by assigning tool motion priorities to avoid collision. This highlights the effectiveness of the proposed approach in improving the fabrication efficiency of biomedical objects.

4.2 A Toy Tank

A tank model with dimensions of 225mm x 78mm x 96mm, as shown in Figure 17, is processed below to demonstrate possible applications of the proposed work in development of complex toy products. The model is sliced into 480 layers. Figure 18 shows a layer of the tank to be made of four materials, represented in black, green, red, and orange. The deposition speeds of the four tools, of the same radius R=10mm, are set to be V_B =20mms^{-1}, V_G =15mms^{-1}, V_O =15mms^{-1}, and V_R=10mms^{-1}, for the black, green, orange, and red tools, respectively.
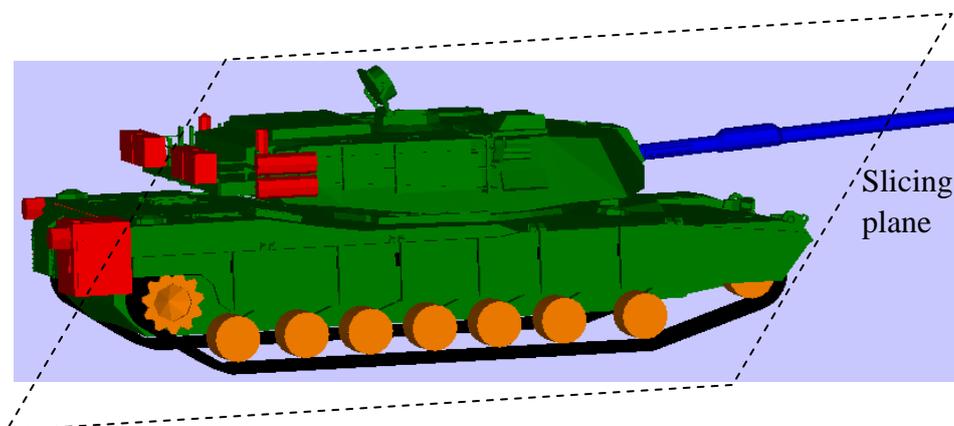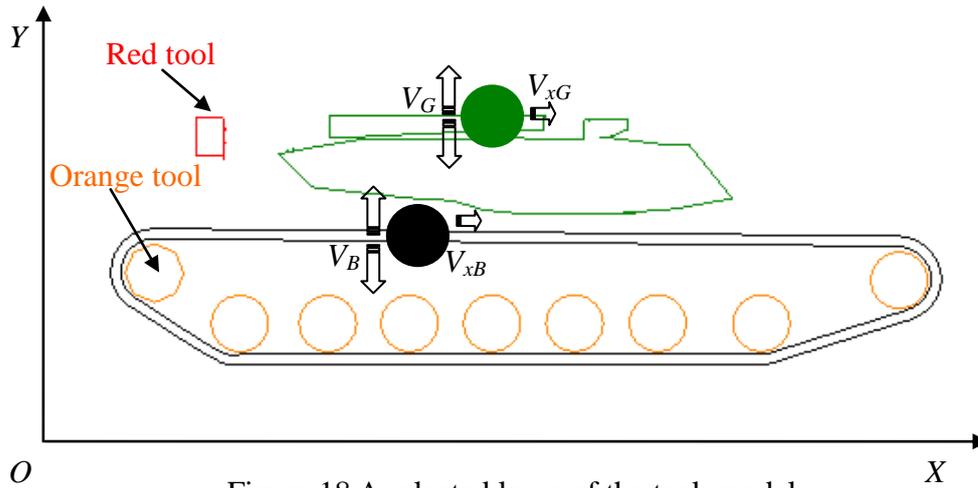

Figure 17 A toy tank model
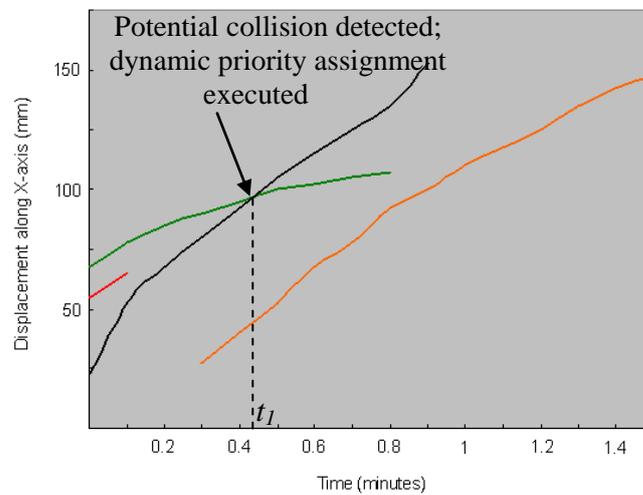
Figure 18 A selected layer of the tank model

Figure 19 *X-t* graph of the selected layer of tank model

At the beginning, since the collision detection algorithm finds that the orange tool will collide with the black tool, the orange tool with lower traverse speed is commanded to start deposition after the potential collision is eliminated. So, at first, the black, red, and green tools move concurrently to deposit their specific materials. Some time later at t_1, a potential collision is detected between the black tool and the green tool, as shown in solid circles in Figure 18 and time t_1 in Figure 19. The motion priorities of the black tool and the green tool are assigned according to their traverse speeds. The black tool, which is at a higher traverse speed, is assigned with a higher priority and thus continues material deposition along its original path together with the orange tool, while the green tool with a lower priority pauses to give way until the potential collision is eliminated. Some time after t_1, the green tool, the black tool, and the orange tool can again deposit materials concurrently to complete this layer. The digital fabrication process of the selected layer of the tank model is shown in Figure 20.

25

(a) Concurrent deposition by black tool, red tool, and green tool

(b) Orange tool starts. Black tool continues to deposit while green tool waits.

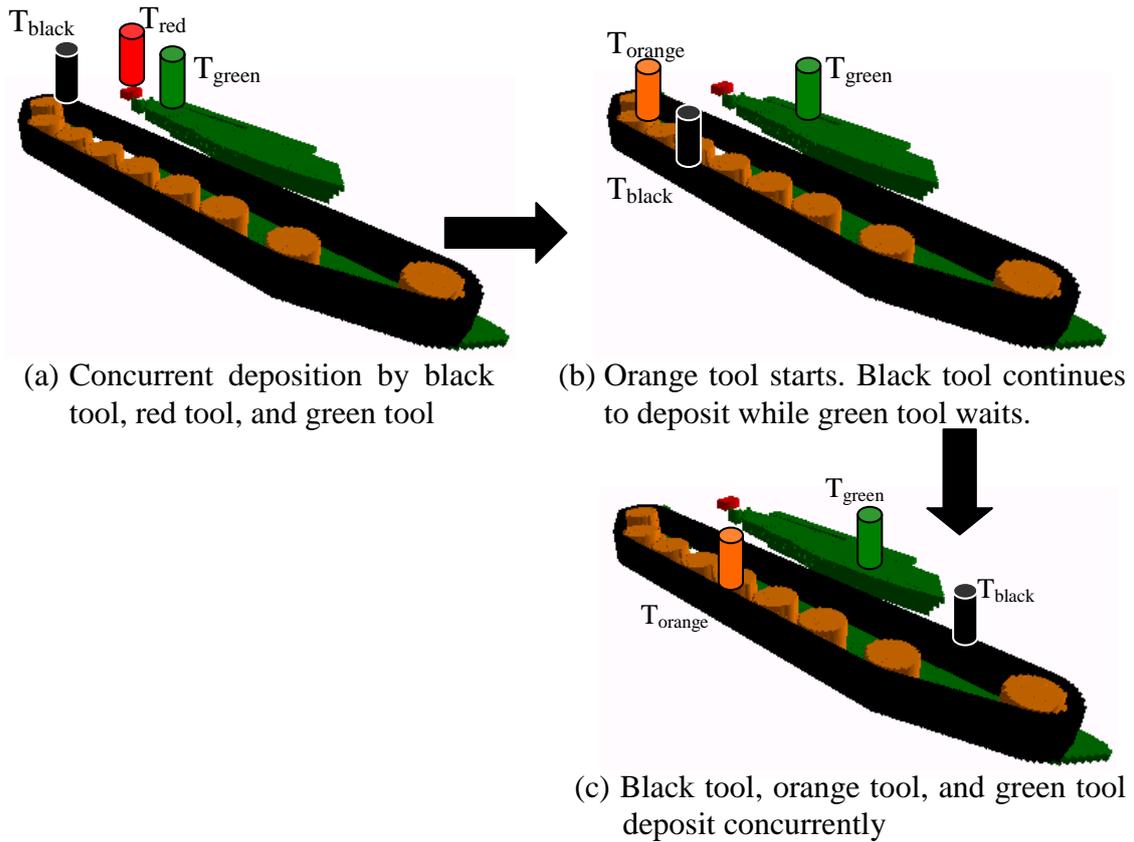(c) Black tool, orange tool, and green tool deposit concurrently

Figure 20 Digital fabrication of the selected layer of the tank model

Figure 21 compares the build times of the tank model by different toolpath planning approaches. It can be seen that the proposed dynamic priority-based approach improves the efficiency by 62%, 51%, and 43% respectively, in comparison with the previous approaches. This highlights that the effectiveness of the proposed approach to improve the fabrication efficiency of relatively complex objects and to shorten the product development cycle accordingly.
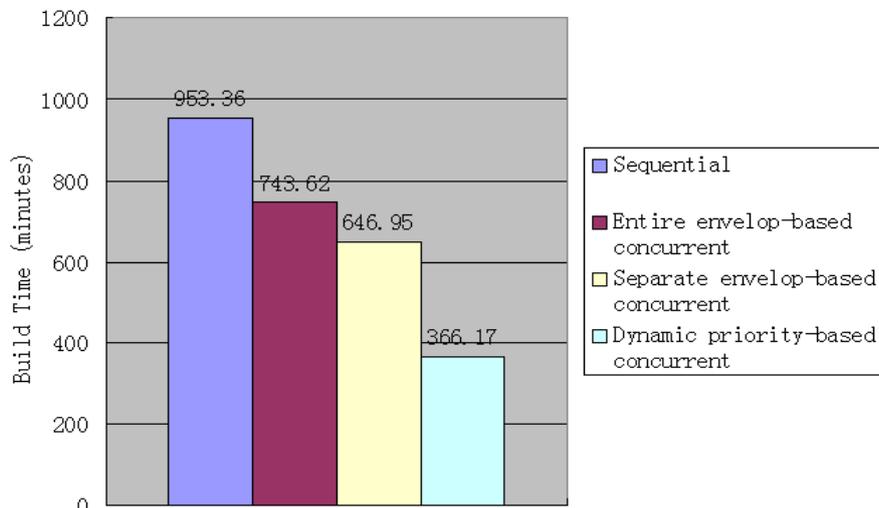

Figure 21 Comparison of build times of the tank model by different toolpath planning approaches

5 Conclusion and Future Work

This paper presents an approach to concurrent toolpath planning to improve the fabrication efficiency of MMLM. The approach incorporates decoupled motion planning technique for multiple moving objects with a collision detection algorithm and a dynamic priority assignment scheme. It is characterised by construction of envelopes directly around individual tools, which are treated as multiple moving objects, while the dynamic priority assignment scheme coordinates the tool motions to avoid collisions. The proposed approach has been integrated with a multi-material virtual prototyping system, and digital fabrication of prototypes for biomedical applications and product development shows that it can substantially shorten the fabrication time of relatively complex multi-material objects. The approach can be adapted for process control of MMLM when appropriate hardware becomes available.

Nevertheless, some further developments are deemed beneficial. Firstly, as the proposed approach now adopts only up-and-down zigzags for internal contour filling, it would be useful to include the spiral contour filling mode, which is also a common contour filling strategy in LM. This may require modifying the algorithms for tool collision detection and motion coordination accordingly.

Secondly, tool collision detection in the proposed approach is a real-time process being executed at a predefined rate. It seems that pausing a tool to avoid collision may possibly impair the smoothness of the fabrication process. Although attempt has been made to reduce such effect by holding the tool only at the far end of the current hatch line after it is completed, it may be worthwhile to further improve the fabrication process. In this connection, it would perhaps be beneficial to take advantage of the X-t graph (tool displacements versus time), which shows the locations and times of possible tool collisions during the complete fabrication process of a layer, to eliminate tool pauses. X-t graphs for all layers would first be generated off-line, from which the locations and times of possible tool collisions could be pre-loaded into the computer for prior coordination of tool motions. For example, the start time of a tool could be suitably adjusted to avoid potential collisions, instead of halting the tools during fabrication. Indeed, this off-line planning would reduce the computation requirements during digital or physical fabrication, particularly for large and complex objects.

Thirdly, the impact of the proposed toolpath planning approach on fabrication quality needs further investigation. Shrinkage and warpage of prototypes may be affected by the adopted toolpath planning strategy. It would be desirable to study the relationship between the toolpath planning strategy and the prototype quality.

6 References

[1] Hopkinson, N., Hague, R.J.M. and Dickens, P.M., Rapid manufacturing: an industrial revolution for the digital age. John Wiley & Sons, Ltd. 2006.

[2] Wohlers, T.T., Wohlers Report 2009, Wohlers Associates, Inc., 2009.

[3] Kulkarni, P., Marsan, A., and Dutta, D., A review of process planning techniques in layered manufacturing, Rapid Prototyping Journal, 2000; 6(1): 18-35.

[4] Kochan, A., Rapid prototyping gains speed, volume and precision. Rapid Prototyping Journal, 2000; 20(4):295-299.

[5] Bellini, A., Fused deposition of ceramics: a comprehensive experimental, analytical and computational study of material behaviour, fabrication process and equipment design, Ph.D. thesis, Drexel University, USA, 2002.

[6] Hauser, C., Dunschen, M., Egan, M. and Sutcliffe, C, Transformation algorithms for image preparation in spiral growth manufacturing (SGM). Rapid Prototyping Journal, 2008; 14(4):188-196.

[7] Sintermask Technologies, http://www.sintermask.com/page.php?p=8, 2007.

[8] Voxeljet, http://www.voxeljet.de/voxeljet_en/inhalt/produkte/3d-drucksysteme/uberblick.php?navid=6 , 2007.

[9] Erasenthiran, P. and Beal, V.E., Functionally graded materials, In: Hopkinson, N., et al., editors. Rapid manufacturing: an industrial revolution for the digital age, John Wiley & Sons, Ltd., 2006; pp.103-124.

[10] Khalil, S., Nam, F. and Sun, W., Multi-nozzle deposition for construction of 3D biopolymer tissue scaffolds, Rapid Prototyping Journal, 2005; 11(1): 9-17.

[11] Kumar, V., Solid modeling and algorithms for heterogeneous objects, Ph.D. thesis, Drexel University, USA, 1999.

[12] Qiu, D., Langrana, N.A., Danforth, S.C., Safari, A. and Jafari, M., Intelligent toolpath for extrusion-based LM process, Rapid Prototyping Journal, 2001; 7(1): 18-23.

[13] Jepson, L.R., Multiple material selective laser sintering, Ph.D. thesis, The University of Texas at Austin, USA, 2002.

[14] Cho, W., Sachs, E.M., Patrikalakis, N.M. and Troxel, D.E., A dithering algorithm for local composition control with three-dimensional printing, Computer-Aided Design, 2003; 35(9): 851-867.

[15] Cesarano III, J., Direct-write fabrication with fine particle suspensions, Advanced Materials and Processes, 2005; 163(8): 49-56.

[16] Inamdar, A., Magana, M., Medina, F., Grajeda, Y. and Wicker R., Development of an automated multiple material stereolithography machine, In: Bourell, D.L. et al., editors. Solid Freeform Fabrication Symposium 2006, University of Texas at Austin, USA, 2006; pp. 624-635.

[17] Wang, J. and Shaw, L.L., Fabrication of functionally graded materials via inkjet

color printing, Journal of American Ceramics Society, 2006; 89(10): 3285-3289.

[18] Malone, E., Berry, M. and Lipson, H., Freeform fabrication and characterization of zinc-air batteries, Rapid Prototyping Journal, 2008; 14(3):128-140.

[19] Objet Geometries Ltd., http://www.objet.com/3D-Printer/Connex350/, 2009.

[20] Choi, S.H. and Cheung, H.H., A topological hierarchy-based approach to toolpath planning for multi-material layered manufacturing, Computer-Aided Design, 2006; 38(2):143-156.

[21] Liang, J.S. and Lin, A.C., System development off adaptive spraying paths for rapid prototyping, Rapid Prototyping Journal, 2003; 9(5):289-300.

[22] Sun, Q., Rizvi, G.M., Bellehumeur, C.T. and Gu, P., Effect of processing conditions on the bonding quality of FDM polymer filaments, Rapid Prototyping Journal, 2008; 14(2): 72-80.

[23] Zhu, W.M. and Yu, K.M., Tool path generation of multi-material assembly for rapid manufacture, Rapid Prototyping Journal, 2002; 8(5):277-283.

[24] Zhou, M.Y., Path Planning of functionally graded material objects for layered manufacturing, International Journal of Production Research, 2004; 42(2):405-415.

[25] Choi, S.H. and Cheung, H.H., A multi-material virtual prototyping system, Computer-Aided Design, 2005; 37(1):123-136.

[26] Choi, S.H. and Zhu, W.K., Efficient concurrent toolpath planning for multi-material layered manufacturing, In: Bourell, D.L. et al., editors. Solid Freeform Fabrication Symposium 2008, University of Texas at Austin, USA, 2008; pp. 429-440.

[27] Ericson, C., Real-Time Collision Detection, Elsevier Inc., 2005.

[28] Jimenez, P., Thomas, F. and Torras C., 3D collision detection: a survey, Computers and Graphics, 2000; 25:269-285.

[29] Laumond, J.P., Kineo CAM: a success story of motion planning algorithm, IEEE Robotics and Automation Magazine, 2006; 13(2):90-93.

[30] Wagner, I.A., Altshuler, Y., Yanovski V. and Bruckstein, A.M., Cooperative cleaners: a study in ant robotics, The International Journal of Robotics Research, 2008; 27(1): 127-151.

[31] Li, T.Y. and Latombe, J.C., On-line manipulation planning for two robot arms in a dynamic environment, The International Journal of Robotics Research, 1997; 16(2):144-167.

[32] Peasgood, M., Clark, C.M. and McPhee, J., A complete and scalable strategy for coordinating multiple robots within roadmaps, IEEE Transactions On Robotics, 2008; 24(2): 283-292.

[33] Lavalle, S.M., Planning algorithms, Cambridge Press, 2006.

[34] Todt, E., Raush, G. and Suarez, R., Analysis and classification of multiple robot coordination methods, In: Proceedings of the IEEE International Conference on Robotics and Automation, San Francisco, California, USA, 2000; pp.3158-3163.

[35] Bennewitz, M., Burgard, W., and Thrun, S., Finding and optimizing solvable priority schemes for decoupled path planning techniques for teams of mobile robots, Robotics and Autonomous Systems, 2002; 41(2-3):89-99.

[36] Hopcroft, J.E., Schwartz, J.T. and Sharir, M., On the complexity of motion planning for multiple independent objects: PSPACE-hardness of the 'warehouseman's problem', International Journal of Robotics Research, 1984; 3(4):76–88.

[37] Lavalle, S.M. and Hutchinson, S.A., Optimal motion planning for multiple robots

having independent goals, IEEE Transactions on Robotics and Automation, 1998; 14(3): 912-925.

[38] Lee, K.H. and Kim, J.H., Multi-robot cooperation-based mobile printer system, Robotics and Autonomous Systems, 2006; 54(3):193-204.

[39] Peng, J. and Akella, S., Coordinating multiple robots with kinodynamic constraints along specified paths, International Journal of Robotics Research, 2005; 24(4): 295–310.

[40] Rekleitist, I., Shuet, V.L, New, A.P. and Choset, H., Limited communication, multi-robot team based coverage, In: Proceedings of International Conference on Robotics and Automation, New Orleans, Louisiana, USA, 2004; pp.3462-3468.

[41] Lee, B.J., Lee, S.O. and Park, G.T., Trajectory generation and motion tracking control for the robot soccer game, In: Proceedings of the IEEE International Conference on Intelligent Robots and Systems, Kyongju, South Korea, 1999; vol.2, pp. 1149-1154.

[42] Chang, C. Chung, M.J. and Lee, B.H., Collision avoidance of two general robot manipulators by minimum delay time, IEEE Transaction On Systems, Man and Cybernetics, 1994; 24(3): 517-522.

[43] van den Berg, J. and Overmars, M., Prioritized motion planning for multiple robots, In: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, 2005; pp: 2217–2222.

[44] Erdmann, M. A. and Lozano-Perez, T., On multiple moving objects, Algorithmica, 1987; 2:477-521.

[45] Ferrari, C., Pagello, E., Ota, J. and Arai, T., Multirobot motion coordination in space and time, Robotics and Autonomous Systems, 1998; 25: 219–229.

[46] Azarm, K. and Schmidt, G., A decentralized approach for the conflict-free motion of multiple mobile robots, In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 1996; pp. 1667–1674.

[47] Clark, C.M., Bretl, T. and Rock, S., Applying kinodynamic randomized motion planning with a dynamic priority system to multi-robot space systems, In: Proceedings of IEEE Aerospace Conference, 2002; vol.7, pp. 3621- 3631.

[48] Choi, S.H. and Cheung, H.H., A versatile virtual prototyping system for rapid product development, Computers in Industry, 2008; 59(5):477-488.