Proceedings of the 1996 IEEE International
Symposium on Intelligent Control
Dearborn, MI • September 15-18, 1996

GENETIC RULE INDUCTION IN THE DESIGN OF COMPUTED-TORQUE/FUZZY-LOGIC CONTROLLERS FOR ROBOTIC MANIPULATORS

B Porter
Department of Industrial and Manufacturing
Systems Engineering
The University of Hong Kong
Pokfulam Road
Hong Kong
e-mail: bporter@hkucc.hku.hk

NN Zadeh
Research Institute for Design, Manufacture,
and Marketing
University of Salford
Salford M5 4WT
England
email: n.n.zadeh@aeromech.salford.ac.uk

ABSTRACT

In this paper, genetic algorithms are used to design computed-torque/fuzzy-logic controllers for robotic manipulators. It is shown that this use of genetic algorithms provides a very effective means of determining the optimal set of fuzzy rules as well as the optimal domains of the associated fuzzy sets of the fuzzy-logic components of such controllers. It is demonstrated that these computed-torque/fuzzy-logic controllers are more robust than computed-torque/fuzzy-logic controllers in which only the domains of the fuzzy sets are genetically determined.

1. INTRODUCTION

In recent years, much research effort has been expended on the development of high-performance trajaectory-tracking controllers for non-redundant robotic manipulators. However, most of the controllers thus developed are too complicated for routine industrial use. Such complicated controllers have therefore failed to replace the relatively simple computed-torque controllers [1] frequently used in practical applications of industrial robots. This circumstance suggests that the following approach to the design of practical high-performance controllers for robotic manipulators be adopted:

(i) retain the well-proven computed-torque structure of such controllers:

(ii) enrich the control algorithms embodied in the computed-torque controllers.

The objective of such a design strategy is to maintain the simplicity of computed-torque controllers whilst increasing their robustness.

The robustness problems associated with conventional computed-torque controllers [1] for robotic manipulators can be readily demonstrated. Thus, if all elastic effects are neglected, the dynamical behaviour of such non-redundant manipulators is governed by non-linear vector-matrix differential equations of the form

$$M(\theta)\ddot{\theta} + h(\theta, \ddot{\theta}) = \tau \ . \tag{1}$$

In equation (1), $M(\theta) \in \Re^{n\times n}$ is the inertia matrix, $h(\theta, \ddot{\theta}) \in \Re^n$ is the vector of centrifugal, coriolis, gravitational, and frictional torques, $\tau \in \Re^n$ is the vector of actuator torques, and $\theta \in \Re^n$ is the vector of joint angles. If all the dynamical characteristics of such manipulators are known precisely, it is possible to use computed-torque/proportional-plus-derivative controllers governed by control-law equations of the form

$$\tau = M(\theta)(\ddot{\theta}_d + u) + h(\theta, \dot{\theta}) \tag{2}$$

where

$$u = K_1 e + K_2 \dot{e} \ . \tag{3}$$

In equations (2) and (3), $\theta_d \in \Re^n$ is the vector of desired joint angles, $e = \theta_d - \theta \in \Re^n$ is the vector of joint-angle errors, $K_1 \in \Re^{n\times n}$ is the proportional gain matrix, and $K_2 \in \Re^{n\times n}$ is the derivative gain matrix. It follows from equations (1), (2), and (3) that

$$\ddot{e} + K_2 \dot{e} + K_1 e = 0 \tag{4}$$

0-7803-2978-3/96/$5.00 © 1996 IEEE

which indicates that the controller gain matrices $K_1 \in \Re^{n\times n}$ and $K_2 \in \Re^{n\times n}$ can be readily chosen so as to produce linear second-order error dynamics with any required characteristics.

However, in practice, the dynamical characteristics of robotic manipulators are rarely known precisely. In such cases, the computed-torque control law equation (2) will have the form

$$\tau = \hat{M}(\theta)(\ddot{\theta}_d + u) + \hat{h}(\theta,\dot{\theta}) \ , \qquad (5)$$

where $\hat{M}(\theta) \neq M(\theta) \in \Re^{n\times n}$, is the estimated value of the inertia matrix and $\hat{h}(\theta) \neq h(\theta) \in \Re^{n}$ is the estimated value of the centrifugal, coriolis, gravitational, and frictional vector. It is evident from equations (2), (3), and (5) that, in such practical cases, the error dynamics are governed by the vector-matrix differential equation

$$\ddot{e} + K_2\dot{e} + K_1 e = \hat{M}^{-1}(\theta)[\{h(\theta,\dot{\theta}) - \hat{h}(\theta,\dot{\theta})\} + \{M(\theta) - \hat{M}(\theta)\}\ddot{\theta}] \ . \qquad (6)$$

Equation (6) indicates that a choice of the controller matrices $K_1 \in \Re^{n\times n}$ and $K_2 \in \Re^{n\times n}$ made on the basis of equation (4) will not normally in practice produce satisfactory error dynamics.

In order to circumvent such robustness problems, adaptive computed-torque controllers have been proposed [1]. However, although these adaptive controllers retain the computed-torque structure, the incorporation of explicit parameter-identification schemes makes such controllers too complicated for routine practical use. In an endeavour to avoid such complications, Porter and Zadeh [2] therefore introduced a new class of computed-torque/fuzzy-logic controllers. Such controllers are non-adaptive but differ from conventional computed-torque/proportional-plus-derivative controllers in that the linear control law (3) is replaced by a non-linear control law of the form

$$u = K(e,\dot{e}) \qquad (7)$$

expressed in terms of fuzzy logic. It was demonstrated by Porter and Zadeh [2] that such computed-torque/fuzzy-logic controllers are more robust than conventional non-fuzzy computed-torque controllers.

However, in these computed-torque/fuzzy-logic controllers of Porter and Zadeh [2], only the domains of the fuzzy sets associated with the fuzzy rules were determined genetically. In the present paper, these results are significantly extended by using genetic algorithms to determine the optimal set of fuzzy rules as well as the optimal domains of the associated fuzzy sets embodied in the computed-torque/fuzzy-logic controllers. It is shown that the resulting computed-torque/fuzzy-logic controllers are more robust than the controllers of Porter and Zadeh [2] (which, in turn, were shown to be more robust than conventional computed-torque/proportional-plus-derivative controllers). It is thus evident that the procedure for genetic rule induction described in this paper is very effective in the design of high-performance controllers for robotic manipulators.

2. GENETIC DESIGN PROCEDURE

The computed-torque/fuzzy-logic controllers under investigation are governed by control-law equation of the forms (2) and (7). The design problem is to express the non-linear controller function, $K(e,\dot{e})$, in equation (7) in terms of fuzzy logic; and the associated design objective is to minimise the cost function

$$\Gamma = \int_0^T \| e(t) \| dt \qquad (8)$$

when the robotic manipulator under control performs a trajectory-tracking task of duration T. This design problem can be solved very readily by using genetic algorithms to determine the optimal set of fuzzy rules as well as the optimal domains of the associated fuzzy sets for such computed-torque/fuzzy-logic controllers.

The simplest form of these controllers is governed by the decoupled version

$$u_i = k_i(e_i,\dot{e}_i) \quad (i = 1,2,\ldots,n) \qquad (9)$$

of equation (7), in which each of the n joints is controlled separately. Thus, each of the n decoupled fuzzy-logic controllers is governed by a rule of the following generic form:

$$\text{If } e_i \text{ is } P_i^{(j)} \text{ and } \dot{e}_i \text{ is } Q_i^{(k)} \ ,$$
$$\text{then } u_i \text{ is } R_i^{(l)} \quad (i = 1,2,\ldots,n) \ . \qquad (10)$$

The entire sets of fuzzy sets in the $e_i, \dot{e}_i$, and u_i spaces are, respectively,

$$P_i = \{P_i^{(1)}, P_i^{(2)}, \ldots P_i^{(p)}\} \quad (i = 1,2,\ldots,n) \ , \qquad (11a)$$

$$Q_i = \{Q_i^{(1)}, Q_i^{(2)}, \ldots Q_i^{(q)}\} \quad (i = 1,2,\ldots,n) \ , \qquad (11b)$$

and

$$R_i = \{R_i^{(1)}, R_i^{(2)}, \ldots R_i^{(r)}\} \quad (i = 1,2,\ldots,n) \ . \qquad (11c)$$

These entire sets of fuzzy sets are symmetric and are, respectively, defined on the domains $[-\alpha_i, +\alpha_i]$, $[-\beta_i, +\beta_i]$, and $[-\gamma_i, +\gamma_i]$, $(i = 1,2,\ldots,n)$.

It is evident from the generic rule (10) that there are altogether (pqn) rules with different antecedents, for each of which the appropriate consequent must be selected from the entire sets $R_i (i = 1,2,\ldots,n)$ of fuzzy sets. In addition, there are 3n domains of these fuzzy sets for each of which the appropriate parameters α_i, β_i, and $\gamma_i \in \Re^+$ $Ii = 1,2,\ldots,n)$ must be determined. In the genetic design procedure, each computed-torque/fuzzy-logic controller governed by equations (2) and (9) is represented by an entire string of (pqn+3n) concatenated sub-strings of binary digits: in each such entire string, the first (pqn) sub-strings represent in encoded form the consequents of the rules, whilst the remaining 3n sub-strings represent in encoded form the domains of these rules. The Darwinian fitness, Φ, of each entire string of binary digits can be conveniently expressed in the form

$$\Phi = \nu - \Gamma \ , \qquad (12)$$

where $\nu \in \Re^+$ is an appropriately large number and the cost function, Γ, is given by equation (8).

The evolutionary process involved in the genetic design of these computed-torque/fuzzy-logic controllers for robotic manipulators starts by generating an initial population of binary strings. Then, using the standard genetic operations of selection, crossover, and mutation [3], successive populations of binary strings are caused to evolve. In this way, computed-torque/fuzzy-logic controllers of progressively increasing fitness are produced. This evolutionary process is allowed to continue until no significant further improvement is achievable, when the binary string representing the controller with the largest achievable fitness, Φ_{max} , is obtained. In view of equation (12), this computed-torque/fuzzy-logic controller has the smallest achievable cost function, Γ_{min} . This evolutionary process thus provides the optimal set of fuzzy rules and the optimal domains of the associated fuzzy sets for the fuzzy-logic components of the required controller. This optimal computed-torque/fuzzy-logic controller minimizes the cost function, Γ, when the robotic manipulator performs the specified task of duration T.

3. ILLUSTRATIVE EXAMPLE

This general methodology for genetic rule induction can be conveniently illustrated by designing a computed-torque/fuzzy-logic controller for the direct-drive two-link robotic manipulator located in the laboratories of the University of Salford. In this case, n=2, p=5, q=5, and r=7. Indeed, in the terminology of equations (11) ,

$$P_i = \{NB, NS, Z, PS, PB\} \quad (i = 1,2) \ , \qquad (13a)$$

$$Q_i = \{NB, NS, Z, PS, PB\} \quad (i = 1,2) \ , \qquad (13c)$$

and

$$R_i = \{NB, NM, NS, Z, PS, PM, PB\} \quad (i = 1,2) \ . \qquad (13c)$$

There are therefore 5 × 5 = 25 rules for each joint of the robotic manipulator. But it is assumed that the set of optimal rules for each joint includes the following rule:

$$\text{If } e_i \text{ is } Z \text{ and } e_i \text{ is } Z \ , \quad \text{then } u_i \text{ is } Z \quad (i = 1,2) \ . \qquad (14)$$

The removal of this rule, and the assumption of symmetry amongst the remaining 24 rules, thus leads to the need to find the appropriate consequent in only 12 rules for each joint of the robotic manipulator.

In each such rule, the possible linguistic values of the constituent fuzzy sub-sets of the sets, $R_i (i = 1,2)$, involved in these consequents (see equation (13c)) can be readily encoded using 3 bits, as shown in Fig 1. It is thus evident that the entire set of consequents of the rules for each computed-torque/fuzzy-logic controller can be expressed by 12×3×2=72 bits. If, in addition, each of the 6 domains is represented by 10 bits, the entire set of domains involved in each such controller can be represented by a further 6×10=60 bits. This means that each computed-torque/fuzzy-logic controller can be represented by a string of 72+60=132 bits.

In applying the genetic algorithm in this case, excellent results were obtained by using a population size N=20, a crossover probability $p_c = 0.8$, a mutation probability $p_m = 0.05$, and by allowing evolution to occur over 500 generations. The fitness of each design was determined by evluating the cost fuction in equation (8) when the robotic manipulator performed a trajectory-tracking task of duration 4.2s with no payload. The resulting genetically induced set of optimal fuzzy rules are shown in Figs 2(a) and 2(b) for the first and second joint of the manipulator, respectively: the genetically determined optimal domains of the associated fuzzy sets are

$$\alpha_1 = 0.00115 \ ,$$
$$\beta_1 = 1.733 \quad , \qquad (15a)$$

$$\gamma_1 = 79.5 \quad .$$

and

$$\alpha_2 = 0.00824 \ , \quad \beta_2 = 2.178 \ , \quad \gamma_2 = 94.8 \quad . \tag{15b}$$

This genetically designed computed-torque/fuzzy-logic controller performs well its trajectory-tracking task, both in the absence of payload and in the presence of a 4kg payload. Indeed, this computed-torque/fuzzy-logic controllers is very robust since only minimal performance degradation occurs as a result of the unmodelled dynamics represented by the payload. In fact, in the absence of payload, the value of the cost function is

$$\Gamma_1 = 20.4 \times 10^{-4} \quad . \tag{16}$$

whereas, in the presence of payload, the value is

$$\Gamma_2 = 45.3 \times 10^{-4} \quad . \tag{17}$$

In the case of the computed-torque/fuzzy-logic controller of Porter and Zadeh [2] (in which only the domains of the associated fuzzy sets were genetically determined), the corresponding values of the cost function are

$$\Gamma_3 = 39.6 \times 10^{-4} \tag{18}$$

and

$$\Gamma_4 = 161.9 \times 10^{-4} \quad . \tag{19}$$

It is evident that $\Gamma_1 < \Gamma_3$ and $\Gamma_2 < \Gamma_4$, thus indicating the superior robustness of the present computed-torque/fuzzy-logic controller (in which the optimal set of fuzzy rules as well as the domain of the associated fuzzy sets are genetically determined).

4. CONCLUSION

In this paper, the results of Porter and Zadeh [2] for the genetic design of computed-torque/fuzzy-logic controllers for robotic manipulators have been significantly extended. In particular, genetic algorithms have been used to determine the optimal set of fuzzy rules as well as the optimal domains of the associated fuzzy sets embodied in these controllers. It has been shown that the resulting computed-torque/fuzzy-logic controllers are more robust than the controllers designed previously by Porter and Zadeh [2].

REFERENCES

[1]J J Craig, "Introduction to Robotics", Addison -Wesley, 1989.

[2]B Porter and N N Zadeh, "Genetic design of computed-torque/fuzzy-logic controllers for robotic manipulators", Proc IEEE International Symposium on Intelligent Control, Monterey, USA August 1995.

[3]D E Goldberg, "Genetic Algorithms in Search, Optimization and Machine Learning", Addison-Wesley, 1989.

Sign Bit	Value Bits	Numeric Value	Fuzzy Value
0	1 1	-3	NB
0	1 0	-2	NM
0	0 1	-1	NS
0,1	0 0	0	Z
1	0 1	+1	PS
1	1 0	+2	PM
1	1 1	+3	PB

Figure 1: Binary representation of the fuzzy consequents

$\dot{e}$ e	NB	NS	Z	PS	PB
NB	NM	Z	Z	PB	NM
NS	Z	PS	NS	NB	Z
Z	NS	NS	Z	PS	PS
PS	Z	PB	PS	NS	Z
PB	PM	NB	PM	Z	PM

Figure 2(a): Genetically induced fuzzy rules: first joint of robotic manipulator

$\dot{e}$ e	NB	NS	Z	PS	PB
NB	Z	PS	NB	PB	Z
NS	PS	NB	NM	Z	PB
Z	NM	NS	Z	PS	PM
PS	NB	Z	PM	PB	NS
PB	Z	NB	PB	NS	Z

Figure 2(b): Genetically induced fuzzy rules: second joint of robotic manipulator