

# 基于基因算法的多变量模糊控制器的设计\*

张毅 李人厚

(西安交通大学系统工程研究所·西安, 710049)

**摘要:** 本文提出了一种通用的模糊控制器的设计方法, 这种方法运用基因算法进行寻优, 具有设计速度快、人工干预少, 可获得一个基于一定性能指标的次优或最优模糊控制器。它可以对多输入多输出(MIMO)系统进行设计, 不需要被控对象的精确数学模型。本文最后以二级倒立摆系统为控制对象给出了一个设计实例和实际控制的结果。

**关键词:** 模糊控制; 基因算法; MIMO 系统

## 1 引言

模糊控制在近几年得到了飞速的发展。这是因为模糊控制器的设计不要建立系统的精确数学模型。它可利用人类现有的语言知识, 具有良好的鲁棒性。现在模糊控制越来越多地应用于工业过程、家用电器等复杂的、不易建模的场合。

但在模糊控制器的设计过程中, 特别是多变量模糊控制器的设计中, 存在以下问题: 第一、模糊控制规则的选取和优化。当系统很复杂或模糊控制输入变量较多时, 模糊规则的可能选取空间急剧增大; 第二、模糊变量的隶属函数的正确选取。在模糊规则确定的情况下, 模糊控制系统的性能由模糊变量的各个子集的隶属函数来决定。这是一个多参数的寻优问题, 在一般情况下无法获得全局最优。Jyhshing, R. Jang 采用神经网络对隶属函数进行优化, 设计了一个次优模糊控制器, 对单级倒立摆进行控制。

目前, 基因算法受到很多学者的重视。它是一种新兴的用于搜索和机器学习的算法。它通过模拟生物进化过程来进行搜索和学习。这种算法具有速度快、可以跳出局部最优等优点, 并越来越多地被用于对许多实际问题的寻优, 如用于神经网络训练等。文献[12]研究了基因算法在控制器设计和调节中的应用, 也把倒立摆系统的控制作为应用实例。本文把基因算法与模糊逻辑结合起来, 利用基因算法对模糊控制器的各参数进行寻优, 实现模糊控制器的自动设计。

本文的内容是这样安排的: 第二节介绍了基因算法的基本原理。第三节讨论在多变量模糊控制器设计过程中存在的问题。第四节介绍基因算法在多变量模糊控制器设计过程中的应用。第五节给出了一个例子, 用基因算法设计一个模糊控制器对二级倒立摆进行控制, 并给出了仿真和实际控制的结果。

## 2 基因算法

基因算法(genetic algorithms)是自然界进化论的数学缩影。自然进化是一个基于群体的优化过程, 通过各种生物优胜劣汰, 适者生存, 最终留下最优秀的个体。基因算法正是通过

\* 国家自然科学基金资助项目。

本文于1995年2月4日收到, 1995年8月23日收到修改稿。

模拟自然进化中的自然选择、基因重组(crossover)和基因突变(mutation)来进行自学习和寻优,是一种随机优化技术,基因算法对复杂的现实问题的寻优效果优于常规算法。

生物的进化是通过染色体的变异和重组实现的,基因算法也模拟这一过程,在算法中染色体是由一串二进制代码组成,其中一位或数位二进制编码构成一个基因,每个基因代表个体的不同属性,个体的总体特性由染色体的各个基因综合决定,基因算法对各个体进行评估,依据适者生存原则留下优秀的个体进行繁殖,基因算法中后代的繁殖分为基因重组和基因突变,基因重组是将二个或二个以上个体的染色体进行重新组合,比如个体 A 染色体中具有以下形式: $x.xaaax.xbbb..x$ ,个体 B 具有: $x.xccc..xddd..x$ 形式,其中  $aaa,bbb,ccc,ddd$  是四个基因,那么个体 A 和个体 B 经基因重组后其后代可具有以下染色体: $x..xccc..xbbb..x$  和  $x..xaaax..xddd..x$ ,一般取总个数的 75%~95%(基因重组率  $C$ )进行基因重组,基因突变是指某一个个体的一个或数个基因发生变异,例如上述的个体 A,其基因  $aaa$  发生突变,它生成的下一代个体具有这样的染色体: $x..xeeex..xbbb..x$ ,基因突变率  $M$  一般为总个体数的 0.05%~15%,基因突变在寻优中非常重要,当寻优陷入局部最优时,可利用基因突变跳出局部最优,基因突变一般有以下种类:二倍体(基因的二进制代码左移 1 位)、半倍体(基因的二进制代码右移 1 位)、突变体(基因的二进制代码重新产生或沿某一方向变化)等等,基因算法的新发展将自然界中的种群迁移、人工筛选等进行模拟,除了  $M, C$  这二个参数之外,在基因算法中还有四个十分重要的参量:即群体规模  $N$ ,代沟  $G$ (控制每代产生期间被代替的个体数),调节范围  $W$  和选择策略  $S$ 。

对于如下寻优问题:

$$\begin{aligned} \min_{x,y,z} J &= g(x,y,z), \\ \text{s.t. } z &= f(x,y) \end{aligned} \quad (1)$$

常规基因算法的求解过程如下:

- 1° 用基因来表示  $x$  和  $y$  的可能取值,即确定调节范围  $W$ 。
- 2° 然后随机初始化一定数目的个体构成初始群体规模  $N$ 。
- 3° 用(1)式建立选择策略  $S$ 。
- 4° 利用选择策略  $S$  和代沟  $G$  对各个体进行评估和择优,留下一定数目的个体。
- 5° 用剩下的个体进行基因重组和基因突变,与父辈一起构成新的个体群。
- 6° 判断寻优是否满足要求,不满足则转到第 3 步,否则转到第 7 步。
- 7° 结束寻优过程。

基因算法可以对搜索空间大和问题复杂等情况进行寻优,在寻优过程中不依赖被求问题的复杂程度,不要求知道被求问题的导数,而且基因算法在初始化个体群、基因重组和基因突变过程中是随机进行的,因此当寻优遇到局部最优时,可通过这种随机过程避免局部最优。

### 3 基于基因算法的模糊控制器的设计

模糊控制器基本结构如图 1 所示,由于 MIMO 的系统可被看成数个 MISO 系统的叠加,因此规则集合一般取如下形式:

$$R_1: \text{ if } x_1 \text{ is } A_1^1 \text{ and } x_2 \text{ is } A_2^1 \cdots \text{ and } x_n \text{ is } A_n^1 \text{ then } u^1 \text{ is } B^1$$

$$\vdots$$

$$(2)$$

$$R_m: \text{ if } x_1 \text{ is } A_1^m \text{ and } x_2 \text{ is } A_2^m \cdots \text{ and } x_n \text{ is } A_n^m \text{ then } u^1 \text{ is } B^m$$

其中  $x_1, x_2, \dots, x_n$ , 表示  $n$  个输入模糊变量,  $A_j^i$  表示第  $j$  个模糊变量的第  $i$  个模糊子集,  $u^j, B^j, j = 1, 2, \dots, m$  表示输出模糊变量及其模糊子集.

实际中如(2)所示的模糊规则集合通常是通过人类语言知识和系统的分析得到的. 规则集合建立后, 模糊控制器还要确定各模糊变量的模糊子集的模糊隶属函数和去模糊化方法. 模糊隶属函数的常见形式有钟形、三角形、梯形等. 以钟形为例, 模糊隶属函数可用

$$\mu_{A_i}(x) = \frac{1}{1 + \left[ \left( \frac{x - c_i}{a_i} \right)^2 \right]^{b_i}}, \quad (3)$$

或

$$\mu_{A_i}(x) = \exp \left\{ - \left[ \left( \frac{x - c_i}{a_i} \right)^2 \right]^{b_i} \right\}. \quad (4)$$

其中  $\{a_i, b_i, c_i\}$  是模糊隶属函数的参数. 去模糊化是将一模糊输出量映射成一个精确输出量, 用作实际控制. 常用的去模糊方法有最大隶属度法、最大隶属度均值法和面积重心法. 在确立了规则、隶属函数和去模糊化方法后, 我们可以建立模糊控制的输入输出关系.

设被控对象由下式表示:

$$\begin{aligned} \dot{x} &= f(x, u), \\ y &= g(x, u), \end{aligned} \quad (5)$$

状态  $\{x_1, x_2, \dots, x_n\}$  为  $n$  维向量, 模糊控制器的输入为状态  $\{x_1, x_2, \dots, x_n\}$ . 依据规则集合(2)中第  $i$  条规则其模糊输出的模糊隶属度为:

$$\mu_{B^i} = \min \{ \mu_{A_1^i}(x_1, a_1^i, b_1^i, c_1^i), \dots, \mu_{A_n^i}(x_n, a_n^i, b_n^i, c_n^i) \}. \quad (6)$$

由(6)可知第  $i$  条规则的模糊输出量的模糊隶属度为  $\{a_j^i, b_j^i, c_j^i | j = 1, 2, \dots, n\}$  的函数, 则整个模糊控制器的模糊输出的模糊隶属度也将是  $\{a_j, b_j, c_j | j = 1, 2, \dots, n\}$  的函数. 以加权平均的方法进行去模糊化, 可得到以下的模糊控制器输入输出关系:

$$u^* = \frac{\sum_{i=1}^m \mu_{B^i}(x, a_i, b_i, c_i) u_i}{\sum_{i=1}^m \mu_{B^i}(x, a_i, b_i, c_i)}. \quad (7)$$

其中  $\mu_{B^i}$  由(6)决定, 从(7)知  $u^*$  由隶属函数的参数  $\{a_i, b_i, c_i\}$  和输入状态  $\{x_1, x_2, \dots, x_n\}$  共同决定. 我们的目的是在输入变量数目确定情况下, 选择隶属函数的参数, 使

$$J = \sum_i (y^i - y_i^d)^2 \quad (8)$$

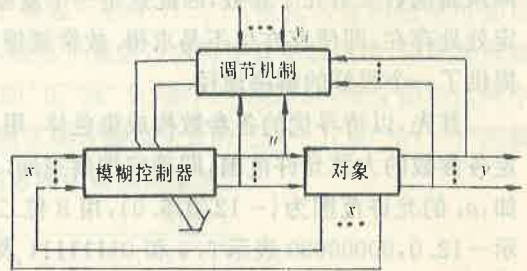


图1 多变量模糊控制结构图



最小, 式中  $y_i$  表示系统实际输出,  $y_i^d$  表示系统期望输出. 这样, 设计一个模糊最优控制器的的问题就成为一个使  $J$  最小, 满足(5)(7) 或(6)(7) 的约束的多维参数寻优问题.

由上述讨论知模糊控制器涉及多个模糊变量, 每个模糊变量有数个子集, 每个子集的模糊隶属函数又有几个参数, 因此这是一个复杂的多维空间搜索问题. 由(6)知  $\mu_{B^i}$  的导数不一定处处存在, 即使存在也不易求得. 故常规搜索方法对这个问题不适用. 而基因算法为我们提供了一个很好的解决途径.

首先, 以待寻优的各参数构成染色体. 用几位二进制数表示一个基因. 利用已知知识确定各参数的大致允许范围, 即确定搜索空间. 然后建立基因与参数之间的一一对应关系. 例如,  $a_1$  的允许范围为  $\{-12.0, 5.0\}$ , 用 8 位二进制数表示  $a_1$  的可能取值, 则有 11111111 表示  $-12.0$ , 00000000 表示  $5.0$  和 01111111 表示  $-3.5$ . 各待求参数的基因共同构成一个染色体, 这个染色体代表一个模糊控制器. 然后, 建立一定的性能指标来衡量模糊控制器的性能, 构成择优目标. 最后, 运用第二节中介绍的基因算法寻优进行求解, 即可以获得一个最优的模糊控制器.

#### 4 运用实例

前述的模糊控制器设计方案设计简单实用, 可用于不同的控制问题. 这里我们将它用于二级倒立摆系统这一典型的控制问题.

##### 1) 二级倒立摆系统.

二级倒立摆系统, 如图 2 所示, 是一典型的非线性复杂系统. 它由一个沿轨道左右运行的小车和沿同一平面自由运动的上、下摆构成. 上摆与下摆、下摆与小车之间以铰链方式联接, 使上摆和下摆都只能在平行于轨道的铅直平面内自由转动. 控制的目标是通过给小车施加一个力, 让小车在轨道上运动以使上摆和下摆保持直立.

二级倒立摆系统的动态特性可以由六个状态变量来描述: 小车位移  $x$ , 小车速度  $\dot{x}$ , 下摆与铅垂线间夹角  $\theta_1$ , 下摆角速度  $\dot{\theta}_1$ , 上摆与铅垂线间夹角  $\theta_2$ , 上摆角速度  $\dot{\theta}_2$ . 对二级倒立摆系统进行分析, 考虑进摩擦的影响后, 可以获得一组微分方程<sup>[11]</sup>.

##### 2) 模糊变量定义与子集划分.

本文中我们采用 Sugeno<sup>[3],[4]</sup> 提出的模糊推理机制其形式:

$$\begin{aligned} R_1: & \text{ if } x_1 \text{ is } A_1^1 \text{ and } x_2 \text{ is } A_2^1 \cdots \text{ and } x_n \text{ is } A_n^1 \\ & \text{ then } u^1 = p_0^1 + p_1^1 x_1 + \cdots + p_n^1 x_n, \\ & \vdots \end{aligned} \quad (9)$$

$$\begin{aligned} R_m: & \text{ if } x_1 \text{ is } A_1^m \text{ and } x_2 \text{ is } A_2^m \cdots \text{ and } x_n \text{ is } A_n^m \\ & \text{ then } u^m = p_0^m + p_1^m x_1 + \cdots + p_n^m x_n. \end{aligned}$$

$\{p_0^1, \dots, p_n^1, \dots, p_n^m\}$  是参数集. 为了简化问题, 不失一般性, 这里假设:

$$\begin{aligned} p_1^1 &= p_1^2 = \cdots = p_1^m = p_1, \\ & \vdots \\ p_n^1 &= p_n^2 = \cdots = p_n^m = p_n. \end{aligned} \quad (10)$$

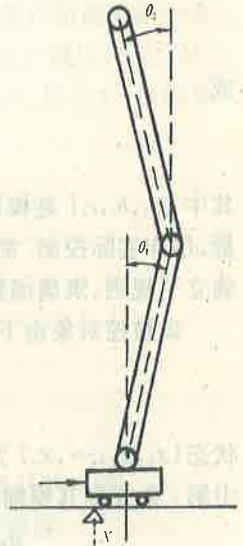


图 2 倒立摆示意图

则参数集简化成  $\{p_0^1, p_0^2, \dots, p_0^m, p_1, p_1, \dots, p_n\}$ . 二级倒立摆系统已有成功的线性控制方案<sup>[1]</sup>, 为了利用已有的成果, 我们作出(10)中的假定简化寻优过程. 我们将前述的六个状态变量定义六个模糊变量, 每个模糊变量划分成二个模糊子集, 则规则库最多可有  $2^6 = 64$  条规则, (9)中  $m = 64, n = 6$ . 每个模糊变量的子集数可以不只二个, 但从后面的结果可知用“正”和“负”二个模糊子集就可实现控制目标, 而且子集数过多将使规则数剧增, 模糊推理减慢, 不利于实时控制. 同时由于倒立摆的方向的“正”或“负”是人为规定的, 倒立摆在这两个方向上不具有特殊性, 因此可作出如下推定: 倒立摆各模糊变量的“正(+)”或“负(-)”两模糊子集的隶属函数左右对称, 假如模糊子集的隶属函数选用下式:

$$\mu_{A_j}(x) \begin{cases} +: \mu_{A_j}(x) = \frac{1}{1 + \left[ \frac{x - c_p}{a_p} \right]^2}^{b_p}, \\ -: \mu_{A_j}(x) = \frac{1}{1 + \left[ \frac{x + c_n}{a_n} \right]^2}^{b_n}, \end{cases} \quad (11)$$

$$a_p = a_n = a, \quad b_p = b_n = b, \quad c_p = -c_n = c.$$

因此每个模糊变量用三个参数即可确定其“正”或“负”两个模糊子集的隶属函数.

### 3) 仿真研究.

由六个模糊变量的隶属函数的参数  $\{a_j, b_j, c_j | j = 1, \dots, 6\}$  和  $\{p_0^1, p_0^2, p_0^m, \dots, p_m^0, p_1, p_2, \dots, p_n | m = 64; n = 6\}$ , 共 88 个构成染色体. 每个基因由一 16 位二进制数表示. 我们选择如下数据作为基因算法中的各参数

$$\begin{array}{ll} \text{群体规模:} & N = 100, \\ \text{基因重组率:} & C = 80\%, \\ \text{基因突变率:} & M = 10\%, \\ \text{代沟:} & G = 30\%. \end{array}$$

系统性能指标为如下标准的二次型指标:

$$J = \sum_{i=0}^N \left( \frac{i^2 \left( \frac{x^2}{10} + \frac{\dot{x}^2}{20} + \theta_1^2 + \dot{\theta}_1^2 + \theta_2^2 + \dot{\theta}_2^2 \right)}{N} \right), \quad (12)$$

以  $J$  越小越好为选择策略  $S$ . 由此在计算机上进行仿真.

图 3 显示了基因算法的寻优结果, 一共生成了 50 代后代. 由图中可见当算到第 25 代以后性能指标  $J$  就已接近最优  $J = 6990.138627$ . 寻优结束后, 各模糊变量的各模糊子集的隶属函数参数如表 1) 所示: 图 4 是用基因算法设计的模糊控制器与同是基于(10)指标设计的次优线性控制器的比较.

在图 4 为了将情况看的更清楚,  $\theta_1, \theta_2, u$  的时间坐标只取了 2 秒, 全部仿真时间为 5 秒. 可看出用基因算法设计的模糊控制器比线性反馈控制器的性能好, 减少了超调, 缩短了过渡时间.

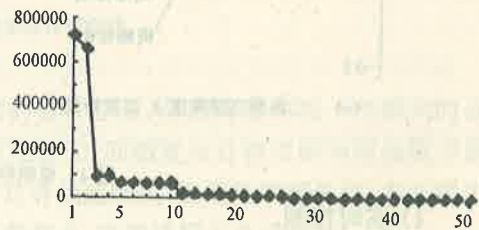
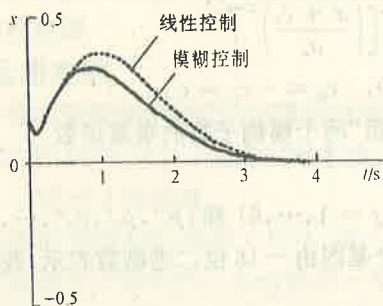


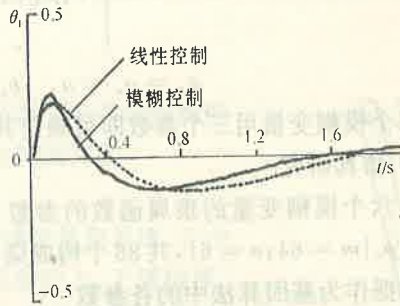
图 3 基因算法性能指标

表 1 各模糊变量的各模糊子集隶属函数参数表

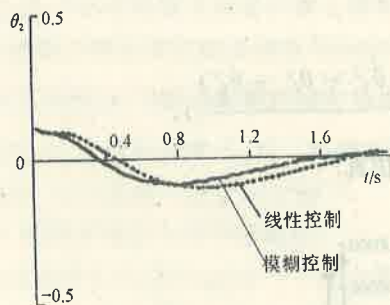
模糊变量	隶 属 函 数			数		
	$a_p$	$b_p$	$c_p$	$a_n$	$b_n$	$c_n$
	正			负		
$x$	10.7470093	0.926084	21.54026	10.747093	0.926084	-21.54026
$\dot{x}$	104.94705	0.660573	49.20957	104.94705	0.660573	-49.20957
$\theta_1$	1.199591	2.532884	0.969543	1.199591	2.532884	-0.969543
$\dot{\theta}_1$	0.544115	3.287759	0.876827	0.544115	3.287759	-0.876827
$\theta_2$	1.114444	1.192969	0.773431	1.114444	1.192969	-0.773431
$\dot{\theta}_2$	1.039003	0.255440	0.949828	1.039003	0.255440	-0.949828



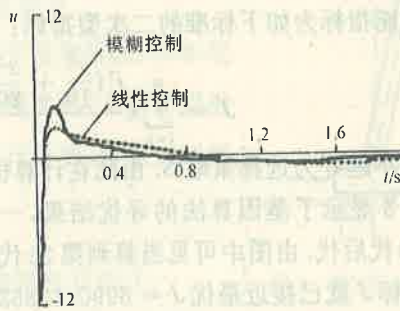
(a) 二级倒立摆位移  $x$  过渡特性



(b) 二级倒立摆角度  $\theta_1$  过渡特性



(c) 二级倒立摆角度  $\theta_2$  过渡特性



(d) 二级倒立摆控制量  $u$  曲线

图 4 模糊控制与线性控制性能比较

4) 实时控制.

我们以第 4 节所得到的模糊控制器对二级倒立摆系统进行实时控制. 使用 12 位的 MS-1215A/D, D/A 板对采样板所输出的小车位移、上下摆角度信号进行采样并将控制器的输出转换为模拟量提供给功放板进行控制. 模糊控制器在 PC386/387 实现, 不记录控制数据时, 控制周期为 8.9ms, 其中模糊推理耗时 6.9ms. 图 5 为一次控制的采样数据.

在采用计算机实现模糊控制对二级倒立摆进行控制时, 主要遇到以下困难: 首先, 由于



采样和模糊推理要花费时间,模糊控制器的实际输出  $u(kT)$  比  $kT$  时刻滞后,这段时间的引入,相当于给闭环系统引入了一个纯时延环节,而二级倒立摆系统是个快速系统,对这种时延特别敏感,需将这种时延尽量减少.最初我们所采用的 PC386 上未带协处理器,所实现的模糊推理器推理一次耗时 70ms,无法实现对二级倒立摆的有效控制.后来插入协处理器 80387,并在推理程序实现上加入一些技巧,使推理时间降为 6.9ms,便可对二级倒立摆系统进行有效控制.

其次,上下摆采样电位器上始终带有 50Hz 的工频干扰.由于这种干扰的存在,要求控制器对上下摆的反馈不宜过大.但是由于现在电机存在较大的死区,又要求控制器对上下摆的反馈不宜太小,因此在选择二次型指标系数时需权衡这两个因素.

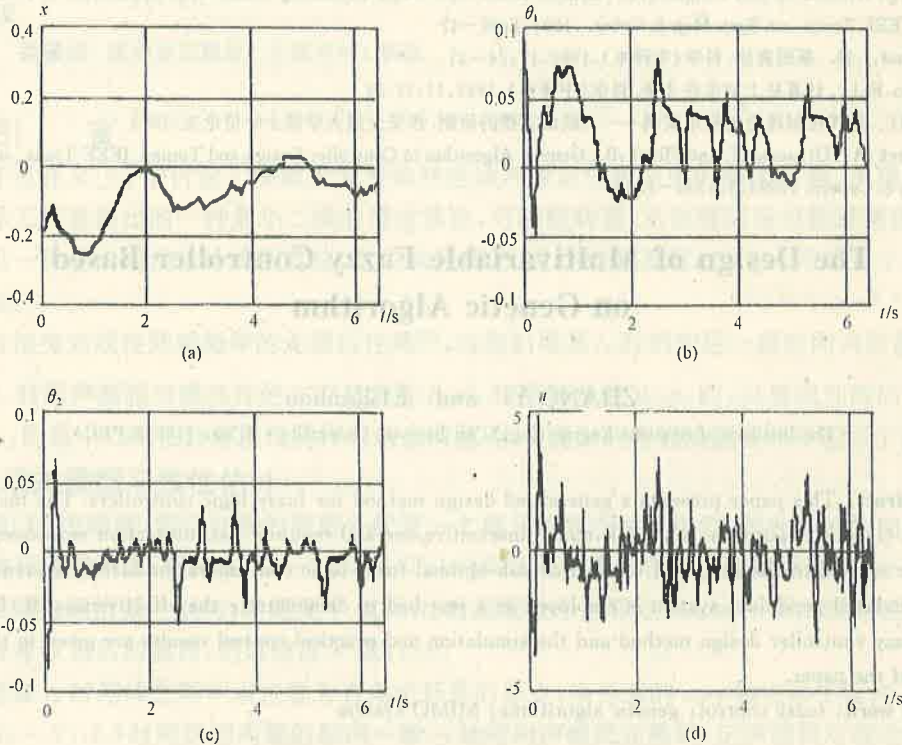


图 5 模糊控制倒立摆实时控制曲线

### 5 结 论

本文提出了一种基于基因算法的模糊控制器的设计方法.基因算法作为一种新兴的寻优和自学习算法,正逐渐受到重视.它对搜索空间维数大、问题复杂且被求解问题导数不易得到的 MIMO 模糊控制器设计问题非常有效,并且寻优速度快,能避免局部最优.本文提出的设计方法具有设计速度快、人工干预少、效果好等优点,非常适用于自动设计.

基因算法这类新方法为模糊控制器的设计提供了一个新的途径,提高了模糊控制器的设计效率,使模糊控制器能更广泛地运用.

### 参 考 文 献

[1] Lee, C. C. . Fuzzy Logic in Control Systems; Fuzzy Logic Controller-Part I. IEEE Trans. on Syst. Man & Cybern. ,

- 1990,2:408-418
- [2] Lee, C. C. . Fuzzy Logic in Control Systems; Fuzzy Logic Controller-Part I. IEEE Trans. on Syst. Man & Cybern. , 1990,2:419-435
- [3] Sugeno, M. . An Introduction Survey of Fuzzy Control. Information Science, 1985,36:59-83
- [4] Sugeno, M. and Nishida, M. . Fuzzy Control of Model Car. Fuzzy Set Systems, 1985,16:103-113
- [5] Jang, J. S. . Self-Learning Fuzzy Controller based on Temporal Back Propagation. IEEE Trans. on Neural Network, 1990,5:714-723
- [6] Potts, J. C. Giddens, T. D. and Yadav, S. B. . The Development and Evaluation of an Improved Genetic Algorithm Based on Migration and Artificial Selection. IEEE Trans. on Syst. Man & Cybern. , 1994,1:73-86
- [7] Fogel, D. B. . An Introduction to Simulated Evolutionary Optimization. IEEE Trans. on Neural Networks, 1994,1:3-14
- [8] Park, D. , Kandel, A. and Langholz, G. . Genetic-Based New Fuzzy Reasoning Model with Application to Fuzzy Control. IEEE Trans. on Syst. Man & Cyber. , 1994,1:39-47
- [9] Holland, J. H. . 基因算法. 科学(中译本), 1992,11:24-31
- [10] Riolo, R. L. . 计算机上的生存斗争. 科学(中译本), 1992,11:77-79
- [11] 冯祖仁. 近代控制理论的应用实例——二级倒立摆的研制. 西安交通大学硕士学位论文, 1982
- [12] Varsek, A. , Urbancic, T. and Filipic, B. . Genetic Algorithm in Controller Design and Tuning. IEEE Trans. on Syst. Man & Cybern. , 1993,5:1330-1339

## The Design of Multivariable Fuzzy Controller Based on Genetic Algorithm

ZHANG Yi and LI Renhou

(The Institute of Systems Engineering, Xi'an Jiaotong University • Xi'an, 710049, PRC)

**Abstract:** This paper presents a generalized design method for fuzzy logic controllers. The method, based on Genetic Algorithm, is a quick, model-insensitive one and requires less interaction with designers. It can be applied to the design of optimal or sub-optimal fuzzy logic controllers for MIMO systems. The double inverted pendulum system is employed as a test-bed to demonstrate the effectiveness of the proposed fuzzy controller design method and the simulation and practical control results are given in the last section of the paper.

**Key word:** fuzzy control; genetic algorithms; MIMO system

### 本文作者简介

**张毅** 1995年毕业于西安交通大学,获硕士学位.现在四川省计经委工作.

**李人厚** 1957年毕业于上海交通大学,现为西安交通大学教授,博士生导师.目前的主要研究方向为系统和控制工程,智能控制理论和应用,著译著作8部,发表论文80多篇.