

基于基因算法的模糊控制研究

蒋海琳, 靳东明

(北京清华大学微电子学研究所, 北京 100084)

摘 要: 本文提出一种基于基因算法优化的自学习模糊控制器的设计. 研究基因算法理论对模糊控制参数的全局寻优. 出于对硬件实现的考虑, 集中讨论了控制规则后件的产生及对规则库的动态学习, 并采用对传统基因算法进行变化后的计算方法. 使用 Matlab 中的仿真工具 Simulink, 对倒立摆典型非线性系统进行了在线模拟, 证实了所提出的控制算法的有效性和适用性, 为进一步的硬件芯片实现的可行性提供了有价值的理论基础.

关键词: 基因算法; 模糊控制; 自学习; 倒立摆

中图分类号: TP309 **文献标识码:** A **文章编号:** 0372-2112 (2002) 05-0676-04

Research on Genetic Algorithms-Based Fuzzy Control

JIANG Hai-lin, JIN Dong-ming

(Institute of Microelectronics, Tsinghua University, Beijing 100084, China)

Abstract: This paper proposes a design of the self-adaptive learning fuzzy controller based on Genetic Algorithms optimization. Researching how this theory optimizes the parameters of fuzzy control in the whole scope is taken. Considering the implementation of hardware, the discussion is focused on the generation of control rules and the dynamic learning of the rules store. Some new computational methods are used based on traditional genetic algorithms. The nonlinear control system of inverted pendulum is simulated on-line with this design, by simulate tools Simulink in Matlab, which proves the validity and the applicability of this proposed control method. This approach also provides a valuable theory basis to the implementation of hardware chip.

Key words: genetic algorithms; fuzzy control; self-adaptive learning; inverted pendulum

1 引言

在复杂的工业控制过程中, 被控对象通常是时变和非线性的, 建立被控对象的精确数学模型较为困难. 解决这类问题较为有效的办法是采用模糊控制技术. 模糊控制可以不依靠被控对象的显式模型, 同时对非线性, 干扰噪声和纯滞后等都有较强的适应能力, 能够实现有效的非线性控制.

但在工程实践中, 一般的模糊控制器存在着自适应能力较差的问题. 其控制规则通常来源人们日常经验的总结固化; 当被控对象的特性随着时间和环境的变化而有较大变化时, 难于总结出完善的控制规则, 模糊控制器无法得到期望的控制效果. 基因算法 (Genetic Algorithms) 通过模拟自然进化中的自然选择、基因重组和基因突变来进行自学习, 可以对大搜索空间和复杂问题等情况进行寻优, 是一种全局随机优化技术. 因此将基因算法用于对模糊控制的优化得到了广泛的研究^[1, 2]. 本文针对已开发的模糊控制芯片, 引入基因算法, 与模糊控制结合, 由基因算法对模糊控制规则进行动态优化, 在线修改, 以实现模糊控制器在线自寻优的控制过程. 并且讨论了该算法在倒立摆典型非线性系统的应用.

2 基因算法

基因算法^[3]是自然界进化论的数学描述, 是一个基于群体的优化过程. 通过各种生物的优胜劣汰, 适者生存, 最终留下最优秀的个体. 基本操作步骤是:

编码 (Encode) 将要优化的参数进行编码, 一般编码为固定长度的二进制形式, 称此二进制串为染色体 (Chromosome);

初始化 (Initialization) 随机生成 N 个染色体;

评价 (Evaluation) 对 N 个染色体进行评价, 评价函数应选取能够客观的反映染色体优劣的函数. 对控制器参数来说, 系统响应的超调量, 稳态误差及响应时间均可反映在评价函数中;

复制 (Reproduction) 随机取出两个染色体作为父本, 适应度越大, 被选中的复制几率越大;

交叉 (Crossover) 将复制后的染色体按一定的交叉概率 P_c 进行基因交换, 交叉位随机选取, 可以是单点交叉, 两点交叉或一致交叉.

变异 (Mutation) 将新产生的染色体以一定的变异概率

P_M 进行基因变异;

判定 (Identification) 反复执行 ~ , 直至达到要求个数的新个体. 它们与父辈构成新的群体. 若达到进化代数或期望要求, 循环结束; 否则跳至 .

3 基于基因算法的模糊控制器

模糊控制器^[4]的设计包括结构和参数两部分. 其中结构是指控制规则的数目、输入输出变量的模糊划分的个数; 参数是指模糊集隶属函数、规则后件等参数.

对于一个多输入单输出系统的模糊控制器, 每一个输入模糊划分的组合一般都与模糊规则相对应, 其模糊规则集 R 由一组“IF—THEN”模糊规则构成:

R_1 : if x_1 is A_{11} and x_2 is A_{12} and x_N is A_{1N} then y_1 is B_1

R_2 : if x_1 is A_{21} and x_2 is A_{22} and x_N is A_{2N} then y_2 is B_2

.....

R_M : if x_1 is A_{M1} and x_2 is A_{M2} and x_N is A_{MN} then y_M is B_M

(1)

其中 x_1, x_2, \dots, x_N 表示 N 个输入变量, A_{ij} 表示第 j 个变量的第 i 个模糊子集, y_i, B_i 表示输出变量及其模糊子集. 规则数目 M 与输入变量个数 N 及相应的模糊划分有关. 输入变量的隶属函数 $\mu_{A_i}(x_i)$ 采用三角形分布形式, 即:

$$\mu_{A_i}(x) = \begin{cases} 0, & x \leq a_i \\ (x - a_i) / (c_i - a_i), & a_i < x \leq c_i \\ (b_i - x) / (b_i - c_i), & c_i < x \leq b_i \\ 0, & b_i < x \end{cases} \quad (2)$$

a_i, b_i, c_i 分别是三角形的三个顶点横坐标.

去模糊化是将一模糊输出量映射成一个精确输出量, 用于实际控制. 模糊控制器的输入状态为 $\{x_1, x_2, \dots, x_N\}$, 第 i 条规则对应的模糊输出的模糊隶属度为:

$$\mu_{B^i} = \min\{\mu_{A_1^i}(x_1, a_1^i, b_1^i, c_1^i), \dots, \mu_{A_N^i}(x_N, a_N^i, b_N^i, c_N^i)\} \quad (3)$$

当规则后件模糊变量采用单点隶属函数 $u = \{u_1, u_2, \dots, u_M\}$, 以重心法去模糊, 控制器的输出为:

$$y = \frac{\sum_{i=1}^M \mu_{B^i}(x, a_i, b_i, c_i) * u_i}{\sum_{i=1}^M \mu_{B^i}(x, a_i, b_i, c_i)} \quad (4)$$

由上式可知, 输出 y 是由隶属函数的参数 $\{a_i, b_i, c_i\}$, 输入状态 $\{x_1, x_2, \dots, x_N\}$, 以及规则后件 u_i 决定的. 优化的目的, 就是调整这些参数, 使得在输入变量数目确定的情况下, 使

$$J = \sum_i (y_i - y_i^d)^2 \quad (5)$$

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.

由于本文研究的最终目的, 是结合硬件实现提出一套可行的优化算法, 因此, 集中研究对规则后件的学习. 因为在我所已开发的模糊控制器的芯片中, 隶属函数的形状固定为三角形, 而规则库则是由 RAM 存放, 可随时读写. 同时, 由于规则库是通过一系列语言规则来表征控制目标和该领域专家的控制策略, 是根据被控系统的行为特性和专家的控制经验写成的, 在没有先验知识或有很少先验知识的情况下, 人为建立规则库是有一定困难的. 并且随着环境的变化也需要更新原

有的规则库以获得更佳的控制效果. 因此, 提出对规则后件的学习是可行的而且必要的.

受 TSK (Takagi-Sugerr-Kang) 模糊控制器^[5]的启发, 规则后件的产生采用对输入变量量化等级的加权来实现. 设输入变量个数为 N , 每个输入的隶属函数的量化等级为 S_i , 则第 k 条规则的后件 u_k 为:

$$u_k = A_{1, s_1} \cdot 1 + A_{2, s_2} \cdot 2 + A_{3, s_3} \cdot 3 + \dots + A_{N, s_N} \cdot N \quad (6)$$

其中 A_{i, s_j} 代表第 i 个输入的第 j 个等级量, $s_1 = 1, 2, \dots, S_1, \dots, s_N = 1, 2, \dots, S_N, \dots$ 是各输入的加权因子. 从物理意义上来看, 就是在控制系统中, 各输入量对系统输出所起到的作用大小的判定. 当算出全部规则后, 应对其进行归一化. 设最大值为 u_{\max} , 最小值为 u_{\min} , 归一化因子为 K ,

$$u_{\text{mid}} = (u_{\max} + u_{\min}) / 2 \quad (7)$$

$$u_i = K \cdot \frac{u_i - u_{\text{mid}}}{u_{\max} - u_{\text{mid}}}$$

这样就把规则库中的 u_i 归一化到 $[-K, K]$ 之间了, 并保证了规则的对称性.

基因算法要学习的参数集即为 $\{1, 2, \dots, N\}$, 个体编码采用级连式多参数编码. 给出每个参数的限定变化范围 (a_i, b_i) , 则参数的实际值与编码后的二进制位串对应关系如下:

$$i = a_i + \left[\sum_{j=1}^k p_j 2^j (b_i - a_i) / (2^k - 1) \right] \quad (8)$$

k 是二进制数的位数, $p_j = 0$ 或 $p_j = 1$.

由于仍然考虑到硬件实现的可能性, 采取以下对传统基因算法稍做变化的方法^[6]. 复制过程, 即父本的选择过程, 采用竞争方式. 随机地从群体中选出 K 个个体来, 每个个体被选中的概率均相等. 然后比较这 K 个个体, 找出其中最好的一个来. 假设群体中共有 n 个个体, 适应度最低的排第 0, 最高的排 $(n-1)$, 对于第 i 个个体, 选中概率 P_i 为

$$P_i = \frac{1}{n} \left(\frac{1}{n} + 2 \cdot \frac{i}{n} \right) \quad (9)$$

可以看到, 每个个体都有被选中的概率, 而且适应度越大, 被选中的概率就越大, 符合基因算法的思想. 通常采用 $K=2$ 的竞争选择, 即每次随机选出两个个体.

每代的交叉操作需要两个父本, 它们应该是独立选择的. 为了简化硬件的设计, 每次只选一个父本, 另一个利用上次产生的. 即每次选出的父本在连续的两代中被利用了两次. 由于基因算法是建立在概率的基础之上的, 这种方法并不影响各个个体被选中的概率, 因此它在理论上是可行的.

交叉方式是一致交叉 (uniform crossover). 它根据一个随机产生的模板来决定后代的各位来自哪个父本.

变异操作对于二进制编码方式, 形式较为单一, 即根据一定的概率将交叉产生的子代中的每位进行翻转. 变异概率对系统的运行有很大的影响. 如果太大, 则将导致群体不收敛; 太小则将导致群体探索新空间的能力不足, 引起“早熟”. 当群体规模较小时, 变异概率应该大一些.

考虑新代如何替换旧代时, 采用稳态 (steady-state) 方式. 每次只产生一个或几个少量的个体. 然后用新个体来替换当

前最差的.

4 GA—Fuzzy 控制在倒立摆系统中的应用

倒立摆系统通常用来检验控制策略的效果,是控制理论界中较为理想的实验手段.倒立摆系统就本身而言,是一不稳定的多变量系统,具有严重的非线性,必须有能够控制的方法使之稳定.图 1 所示是一阶倒立摆的物理模型,一个带轮小车,其顶端用铰链系一刚性杆,小车可沿一笔直的有界轨迹自由的向左或向右运动,同时刚性杆可在垂直平面自由运动.在任一时刻 t ,该系统的状态可由四个实值变量来描述,即小车位置 x ,小车平移速度 \dot{x} ,杆偏离垂直方向的角度 θ ,以及杆的角速度 $\dot{\theta}$.一阶倒立摆的微分方程为:

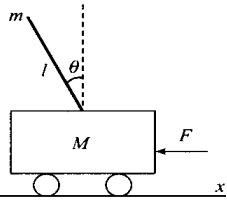


图 1 一阶倒立摆示意图

$$\ddot{\theta} = \frac{g \sin \theta + \cos \theta \left(\frac{-F - m l \dot{\theta}^2 \sin \theta}{M + m} \right)}{l \left(\frac{4}{3} - \frac{m \cos^2 \theta}{M + m} \right)} \quad (10)$$

$$\ddot{x} = \frac{F + m l (\dot{\theta}^2 \sin \theta - \ddot{\theta} \cos \theta)}{M + m}$$

控制的目的是通过给小车底座施加一个力 F ,在尽可能短的时间内将小车从初始位置稳定到给定的零位移处,且使杆在小扰动的情况下仍能保持直立不倒.

以倒立摆为控制对象的 GA-Fuzzy 控制系统如图 2 所示.控制器的四个输入分别为倒立摆的系统状态值 $\theta, \dot{\theta}, x, \dot{x}$,经过模糊算法得到小车的控制力 F .

将每个输入变量的隶属度函数都划分为三个等级,称之为 N (负), Z (零), P (正),采用三角形隶属函数.输入的加权因子分别为 w_1, w_2, w_3, w_4 ,相应的变化范围是:

$$9 \leq \theta \leq 11, 7 \leq \dot{\theta} \leq 9, 5 \leq x \leq 7, 3 \leq \dot{x} \leq 5, \quad (11)$$

也就是说,我们认为在控制过程中 θ 和 $\dot{\theta}$ 起到的作用较大,即控制使摆不倒下是首要任务,而位置参量为次要考虑因素,所以加权较小.(这里面的数字只反映了相互之间的比例关系,即以 10 8 6 4 为中心的一个比例范围.因为我们最终使用的规则库中的规则后件,将由归一化因子 K 进行归一化;与倒立摆物理模型中相对应的力的大小,由 K 来决定.)每个参数由八位二进制数构成,这样得到的染色体的长度为 32 位.基因算法的主要参数是:种群大小 16,交叉概率 0.9,变异概率 0.1.具体操作过程如下:

GA 随机产生一组(16 个)二进制基因串.

按照实际值与编码值之间的关系,计算出参数的实际值.

按前文阐述的方法,计算出相应的规则后件.

输入根据每一套规则库经过模糊推理和去模糊后,计算出控制力 F 作用在倒立摆系统上,得到四个状态输入,作为该套规则控制的评价标准;同时作为下次模糊控制的输入量.适应度函数取:

$$f = \left[\theta^2 + \dot{\theta}^2 + \frac{x^2}{20} + \frac{\dot{x}^2}{10} \right] \quad (12)$$

出于对控制中状态的突变产生的影响,最终选择 f 作为评价函数,即:

$$f(i) = f(i) - f(i-1) = [\theta(i)^2 + \dot{\theta}(i)^2 + x(i)^2/20 + \dot{x}(i)/10] - [\theta(i-1)^2 + \dot{\theta}(i-1)^2 + x(i-1)^2/20 + \dot{x}(i-1)/10] \quad (13)$$

经过一代的控制后,在个体中选择优良个体,进行交叉,变异等基因操作,产生一个新个体,来代替本代中最差的个体,跳至 ,开始下一代的寻优,直至系统稳定.

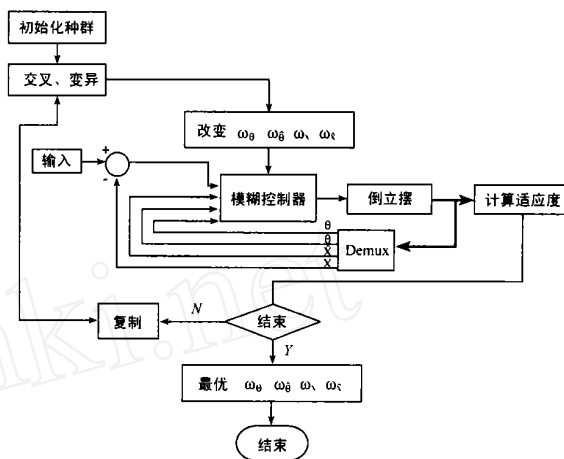


图 2 基因优化模糊控制倒立摆流程

用 Matlab 中的 Simulink 工具^[7]对倒立摆进行了实时控制,GA-Fuzzy 算法在 S-Function 中实现.小车质量 $M = 1.0\text{kg}$,摆质量 $m = 0.1\text{kg}$,摆长度 $l = 1.0\text{m}$,重力加速度 $g = 9.8\text{m/s}^2$.小车的初始位置为 $x = 1.0\text{m}$,倒立摆的初始角度为 $\theta = 0\text{rad}$.仿真时间为 30s,采样步长为 0.01s.图 3、图 4 分别是小车位置和倒立摆夹角的仿真曲线.其中实线是使用 GA 算法在线优化后的模糊控制曲线,而虚线是没有使用 GA 算法的模糊控制曲线.

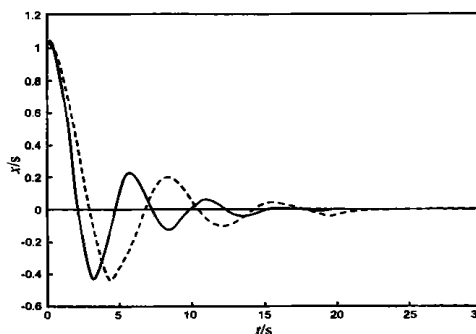


图 3 小车位置的仿真曲线

5 结论

从仿真结果可以看出,在使用了基因算法的在线优化后,收敛速度加快,超调减小.基因算法作为一种寻优和自学习方法,对搜索空间维数大,问题复杂且被求问题导数不易得到的 MMO 模糊控制器的设计问题非常有效,并且人工干预少,能

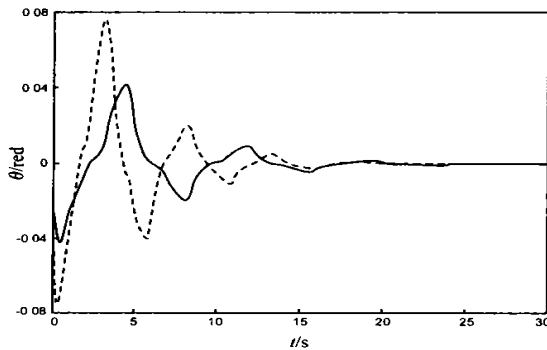


图 4 倒立摆与垂直夹角的仿真曲线

避免陷入局部最优,适用于自动化设计.同时,我们所提出的算法均是适用于硬件实现的,因此在今后的同类产品的芯片开发中,为进一步的硬件实现的可行性提供了有价值的理论基础,将会大大加快开发速度.

参考文献:

- [1] Li Tzur-Hseng S, Shieh Ming-yuan. Design of a GA-based fuzzy PID controller for non-minimum phase systems [J]. Elsevier Science Publishers B V, Fuzzy Sets and Systems, 2000, 111 (2) : 183 - 197.
- [2] Zhou Yi-sheng, Lai Lin-Ying. Optimal design for fuzzy controllers by genetic algorithms [J]. IEEE Transactions on Industry Applications 36 1 Jan-Feb 2000 IEEE:93 - 97.
- [3] Francisco Herrera, Jose Luis Verdegay. Genetic Algorithms and Soft Computing [M]. Heidelberg: Physica Verlag, 1996. 3 - 25.
- [4] Pedrycz, Witold. Fuzzy Control and Fuzzy Systems [M]. Taunton, Somerset, England: Research Studies Press, 1993. 3 - 35.
- [5] Passino, Kevin. Fuzzy Control [M]. Menlo Park, Calif: Addison-Wesley, c1998. 69 - 73.
- [6] 钟国安, 靳东明. 基于硬件实现的基因算法的研究 [J]. 电子学报, 2000, 28(11) : 72 - 76.
- [7] Dabney James. Mastering SIMULINK Z [M]. James B Dabney, Thomas L Harman, Upper Saddle River, NJ: Prentice Hall, 1998. 80 - 120.

作者简介:



蒋海琳 女, 1976 年 2 月出生于北京市, 1999 年毕业于清华大学电子工程系, 获工学学士学位, 并直读清华大学微电子所硕士, 2001 年获清华大学工学硕士学位, 研究方向是基于硬件实现的基因算法用于模糊控制的优化研究.



靳东明 男, 1942 年 2 月出生于天津市, 清华大学微电子所教授、博士生导师, 1965 年毕业于清华大学无线电电子学系, 并留校任教, 目前在多值逻辑、模糊控制和神经网络的硬件实现方面开展研究工作, 现为中国电子学会高级会员, 中国计算机学会多值逻辑与模糊逻辑专业委员会委员.