

文章编号: 1000-8152(2006)01-0001-06

支持向量机-模糊推理自学习控制器设计

袁小芳, 王耀南, 孙 炜

(湖南大学 电气与信息工程学院, 湖南 长沙 410082)

摘要: 常规的模糊推理系统大多由专家经验建立模糊规则, 自学习能力不强. 提出了一种支持向量机-模糊推理系统, 由支持向量机实现模糊推理系统的自学习, 并设计了一种支持向量机-模糊推理自学习控制器. 文章给出了自学习控制器的结构和学习算法, 对比研究了变尺度梯度优化和混沌优化两种学习算法. 针对非线性对象的仿真实验验证了该控制器的优良性能, 控制效果比模糊逻辑控制器更好.

关键词: 模糊逻辑; 模糊推理系统; 支持向量机; 自学习

中图分类号: TP18 **文献标识码:** A

Self-learning controller using support vector machines and fuzzy inference system

YUAN Xiao-fang, WANG Yao-nan, SUN Wei

(College of Electrical and Information Engineering, Hunan University, Changsha Hunan 410082, China)

Abstract: As conventional fuzzy inference system (FIS) was derived from expert experience, it has poor ability in self-learning or adaptation. The self-learning capability of fuzzy inference system was realized in this paper using support vector machines (SVM), and a self-learning controller based on support vector machines-fuzzy inference system (SVM-FIS) was proposed. Both the structure and learning algorithms of the proposed self-learning controller were analyzed. Two learning algorithms of Multi-scaled Davidon-Fletcher-Powell (MDFP) method and chaotic optimization were compared. Simulation results for a nonlinear system demonstrate that the proposed self-learning controller has better control performance over fuzzy logic controller.

Key words: fuzzy logic; fuzzy inference system; support vector machines (SVM); self-learning

1 引言 (Introduction)

模糊推理系统 (Fuzzy Inference System, FIS) 是建立在模糊集合理论、模糊 if-then 规则和模糊推理等概念基础上的计算框架, 在自动控制、专家系统、模式识别等多个领域取得了不少成功的应用, 在控制领域又称为模糊逻辑控制器^[1]. 为了增强模糊推理系统的自学习和自适应能力, 一些学者研究了将神经网络与模糊推理系统结合的神经模糊系统^[2], 利用神经网络来实现模糊推理系统的自学习. 由于神经网络是基于经验风险最小化准则, 因而存在过学习、局部极小点、结构设计依赖于专家经验等缺点.

支持向量机 (Support Vector Machines, SVM)^[3] 是 Vapnik 等提出的一种新的机器学习算法, 与神经网络不同, 支持向量机是依据结构风险最

小化准则, 具有小样本学习、全局最优、泛化能力强等特点^[3]. 本文提出了一种支持向量机-模糊推理系统 (SVM-FIS), 由支持向量机实现模糊推理系统的自学习, 并设计了一种自学习控制器. 文章给出了自学习控制器的结构和学习算法, 自学习控制器包括支持向量机-模糊推理系统、支持向量机辨识器 (SVMM) 两个部分, 分别用于控制决策和系统辨识. 在学习算法方面, 对比研究了变尺度梯度优化和自适应混沌优化两种不同的学习算法.

自学习控制器既有模糊推理系统不依赖对象模型、鲁棒性强的特点, 又有支持向量机学习泛化能力强的特点, 针对非线性对象的仿真实验验证了该控制器的优良控制性能, 控制效果比模糊逻辑控制器更好.

收稿日期: 2004-08-09; 收修稿日期: 2005-04-20.

基金项目: 国家自然科学基金资助项目 (60375001); 高校博士点基金资助项目 (20030532004).

2 支持向量机-模糊推理系统 (Support vector machines-fuzzy inference system)

常规的模糊推理系统包括:规则库,包含一系列模糊规则;数据库,定义模糊规则中用到的隶属函数;推理机制,按照规则和所给事实执行推理过程求出合理的输出或结论. 当有精确输入和输出时,模糊推理系统实现从输入到输出之间的非线性映射. 在控制领域中,模糊推理系统一般是两输入(误差 E 和误差变化率 EC) 单输出(控制量 U). 建立模糊集论域 $\{NB, NM, NS, Z, PS, PM, PB\}$, 对应7个等级 $\{-3, -2, -1, 0, 1, 2, 3\}$, 以 Mamdani 模糊模型为例, 一个具有49条规则的两输入(E, EC)、单输出(U) 模糊控制规则库可以表示为:

规则1: IF E is NB and EC is NB THEN U is PB;

规则2: IF E is NM and EC is NB THEN U is PM;

...

规则49: IF E is PB and EC is PB THEN U is NB.

对于 E, EC 及 U 选取合适的隶属函数, 使用极大-较小复合及中心去模糊化策略, 可以得到如图1所示的模糊推理系统输入-输出曲面.

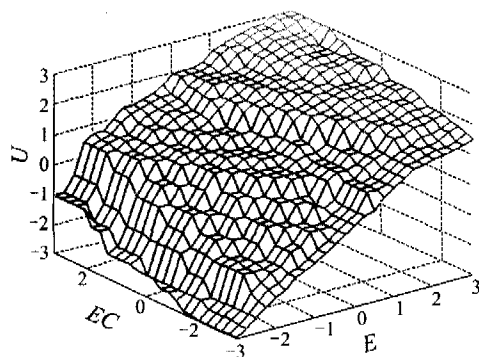


图1 二维模糊控制器的输入-输出曲面

Fig. 1 Input-output relation of fuzzy logic controller

这里的支持向量机为回归形式, 又称支持向量回归 (Support Vector Regression, SVR), 具体描述见文献[3,4]. 支持向量回归具有很强的非线性逼近能力, 其原理是用一个非线性映射将数据映射到高维特征空间, 再在高维特征空间进行线性回归. 支持向量机-模糊推理系统就是利用支持向量回归的非线性逼近能力去逼近模糊推理系统输入-输出曲面, 再以优化支持向量机参数的方式来实现模糊推

理系统的自学习.

由图1所示的输入-输出曲面建立学习样本集 $\{((E_1, EC_1), U_1), \dots, ((E_n, EC_n), U_n)\} \subset \mathbb{R}^2 \times \mathbb{R}$, 支持向量回归的学习过程转化为一个二次规划问题, 学习结果得到支持向量 $((E_i, EC_i), U_i)$ 和对应的系数 $(\alpha_i^* - \alpha_i)$, 用 \vec{E}_i 表示 (E_i, EC_i) , 从而得到回归函数为

$$f(\vec{E}) = \sum_{i=1}^g (\alpha_i^* - \alpha_i) K(\vec{E}, \vec{E}_i) + b. \quad (1)$$

这里 g 表示支持向量个数, 用 w_i 表示 $(\alpha_i^* - \alpha_i)$, 将 b 看作一个权值 w_{g+1} , 采用 RBF 核函数: $K(x_i, x_j) = \exp(-|x_i - x_j|^2 / \sigma^2)$, 有 $K(\vec{E}, \vec{E}) = 1$, b 相当于增加一个支持向量 \vec{E} , 方程(1)可表示为

$$f(\vec{E}) = \sum_{i=1}^{g+1} w_i K(\vec{E}, \vec{E}_i) = \sum_{i=1}^{g+1} w_i \exp(-|\vec{E} - \vec{E}_i|^2 / \sigma^2). \quad (2)$$

支持向量机在结构上为一个3层网络, 在这里输入层节点有2个, 对应于 E, EC , 隐层节点数目就是支持向量数目, 即每一个隐层节点对应一个支持向量, 输出层节点1个, 即 U . 输入层与隐层的连接权值为1, 隐层与输出层的连接权值为 $(\alpha_i^* - \alpha_i)$.

3 自学习控制器的结构 (The structure of self-learning controller)

自学习控制器包括两个部分: 支持向量机-模糊推理系统 (SVM-FIS) 和支持向量机辨识器 (SVM-M), 其结构如图2所示. 支持向量机-模糊推理系统用于控制决策, 输入为二维 (E 和 EC), 其中间部分为支持向量机, 输出为方程(2)所示的回归函数. 支持向量机-模糊推理系统由支持向量机来逼近模糊推理系统的输入-输出曲面, 再以优化支持向量机参数的方式来实现模糊推理系统的自学习能力. 支持向量机辨识器是由支持向量机来辨识系统模型, 通过不断获得对象输入输出数据, 由回归算法在线建立对象模型, 并可提供支持向量机-模糊推理系统学习过程中的信号 $\frac{\partial y_k}{\partial u_k}$.

4 自学习控制器的学习算法 (Learning algorithms for self-learning controller)

常规的模糊推理系统是依据专家经验来设定模糊规则, 因而图1所示的输入-输出曲面需要学习优化, 其优化可通过调整支持向量机参数来实现. 实际控制中对于不同的控制对象, 比例因子参数 K_p, K_c

和 K_u 也需要相应的整定. 因而自学习控制器的学习包括调整支持向量机网络权值和比例因子参数, 这里对比研究了变尺度梯度优化和自适应混沌优化两种学习算法.

定义学习优化的性能指标为

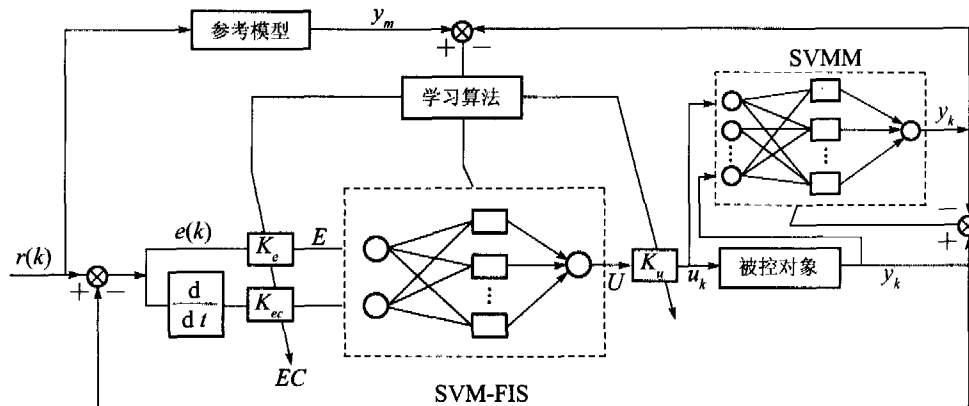


图 2 自学习控制器结构

Fig. 2 Structure of the proposed self-learning controller

4.1 变尺度梯度优化学习算法 (Multi-scaled Davidon-Fletcher-Powell (DFP) learning algorithms)

4.1.1 变尺度梯度优化算法 (Multi-scaled DFP method)

变尺度梯度优化 (MDFP) 算法的基本思想是, 在极小点附近用二阶 Taylor 多项式近似目标函数 $J(W, K)$, 进而求出极小点的估计值. 算法的具体推导过程参考文献 [5], 可知支持向量机网络权值 W 和比例因子参数 K 的递推公式为:

$$\begin{cases} W_{k+1} = W_k - H_k E_k(W_k) \cdot \nabla E_k(W_k) / \beta_k, \\ K_{k+1} = K_k - H_k E_k(K_k) \cdot \nabla E_k(K_k) / \beta_k, \\ H_{k+1} = \lambda^{-1} (H_k - H_k \nabla E_k \cdot \nabla E_k^T \cdot H_k / \beta_k), \\ \beta_{k+1} = \lambda + \nabla E_k^T \cdot H_k \cdot \nabla E_k. \end{cases} \quad (4)$$

这里 $0 < \lambda < 1$ 为遗忘因子, H 为 Hessian 矩阵且 $H_1 = I$ (单位阵), $E_k = (y_m - y_k)$, ∇E_k 分别表示 $\nabla E(K_k)$, $\nabla E(W_k)$ 的梯度. 由于 H_k 是正定的, 从式 (4) 中可以看出, H_{k+1} 也是正定阵, β_k 总是正数, 修正公式 W_{k+1} 和 K_{k+1} 总是沿着 $E(W_k)$, $E(K_k)$ 的负梯度方向一致收敛, 从而保证学习算法的收敛性.

下面计算网络权值的梯度:

$$\begin{aligned} \nabla E_k(W_k) &= \frac{\partial J}{\partial W_k} = \frac{\partial J}{\partial y_k} \cdot \frac{\partial y_k}{\partial u_k} \cdot \frac{\partial u_k}{\partial W_k} = \\ &= -(y_m - y_k) \cdot \frac{\partial y_k}{\partial u_k} \cdot K_{u_k} \cdot K(\vec{E}, \vec{E}_k). \end{aligned} \quad (5)$$

而比例因子参数的梯度为

$$J(W, K) = \frac{1}{2} \sum_{i=1}^M E_i^2(W, K) = \frac{1}{2} \sum_{k=1}^M (y_m - y_k)^2. \quad (3)$$

其中向量 W 为网络权值, 向量 K 为比例因子参数 K_e , K_{ec} 及 K_u , E_i 为系统输出 y_k 与期望输出 y_m 的误差.

$$\nabla E_k(K_k) = \frac{\partial J}{\partial K_k} = \frac{\partial J}{\partial y_k} \cdot \frac{\partial y_k}{\partial u_k} \cdot \frac{\partial u_k}{\partial K_k}. \quad (6)$$

对于 K_u :

$$\begin{aligned} \nabla E_k(K_{u_k}) &= -(y_m - y_k) \cdot \frac{\partial y_k}{\partial u_k} \cdot \\ &= \sum_{i=1}^{g+1} w_i \exp\left(-\frac{|\vec{E} - \vec{E}_i|^2}{\sigma^2}\right). \end{aligned} \quad (7)$$

对于 K_e 和 K_{ec} :

$$\begin{aligned} \nabla E_k(K_{e_k}) &= -(y_m - y_k) \cdot \frac{\partial y_k}{\partial u_k} \cdot K_{u_k} \cdot \\ &= \sum_{i=1}^{g+1} w_i \exp\left(-\frac{|\vec{E} - \vec{E}_i|^2}{\sigma^2}\right) \left(-\frac{2E_{k-1}}{\sigma^2}\right), \end{aligned} \quad (8)$$

$$\begin{aligned} \nabla E_k(K_{ec_k}) &= -(y_m - y_k) \cdot \frac{\partial y_k}{\partial u_k} \cdot K_{u_k} \cdot \\ &= \sum_{i=1}^{g+1} w_i \exp\left(-\frac{|\vec{E} - \vec{E}_i|^2}{\sigma^2}\right) \left(-\frac{2(E_{k-1} - E_{k-2})}{\sigma^2 T}\right). \end{aligned} \quad (9)$$

在式 (5) ~ (9) 中, \vec{E}_i 仍为 (E_i, EC_i) , $E_k = (y_m - y_k)$, T 为采样周期, $\frac{\partial y_k}{\partial u_k}$ 由系统辨识反传求得.

4.1.2 支持向量机辨识器 (Support vector machines modelling)

考虑单输入单输出非线性系统:

$$y(k+1) = f(y(k), \dots, y(k-n), u(k), \dots, u(k-m)), m \leq n. \quad (10)$$

其中 u 为控制输入, y 为系统输出. 设有连续的控制

输入 $u(k-m), u(k-m+1), \dots, u(k)$ (当 $k-m+r < 0$ 时, $u(k-m+r) = u(0), r = 0, 1, \dots, m$) 和系统输出 $y(k-n), y(k-n+1), \dots, u(k)$ (当 $k-n+r < 0$ 时, $y(k-n+r) = y(0), r = 0, 1, \dots, n$), 则对应的输出为 $y(k+1)$.

设 $x(i) = (y(i), \dots, y(i-n), u(i), \dots, u(i-m))$, 则

$$y(i+1) = f(x(i)). \quad (11)$$

通过不断获得新的样本, 构造学习样本集 $(x(i), y(i+1))$, 由支持向量回归建立对象模型:

$$\hat{y}(k+1) = \sum_{i=1}^g (\alpha_i^* - \alpha_i) K(x(k), x(i)) + b. \quad (12)$$

为了动态地辨识系统模型, 支持向量机辨识器 (SVMM) 采用一种在线支持向量回归 (AOSVR)^[6] 方法在线建模. 在线支持向量回归方法采用递推式学习算法, 在样本集合 T 的基础上增加一个新样本 $(x(c), y(c+1))$ 时, 就逐步修改各个样本的系数, 直到所有样本都满足 Karush-Kuhn-Tucker (KKT) 条件时, 得到更新的估计模型 $\hat{y}(k+1)$. 设偏差函数 $e(i) = \hat{y}(i+1) - y(i+1)$, 定义关于 $e(i)$ 的 3 个集合 E, S, R , 每个样本都属于这 3 个集合的唯一一个:

- 1) Error support vectors: $E = \{i \mid |w_i| = C\}, |e(i)| > \varepsilon$;
- 2) Margin support vectors: $S = \{i \mid 0 < |w_i| < C\}, |e(i)| = \varepsilon$;
- 3) Remaining samples: $R = \{i \mid w_i = 0\}, |e(i)| < \varepsilon$.

当增加新样本 $(x(c), y(c+1))$ 时, 在线支持向量回归方法的步骤如下:

Step 1 w_c 取初始值 0, 由 (11) 计算 $\hat{y}(k+1)$, 并计算 $e(c) = \hat{y}(c+1) - y(c+1)$;

Step 2 若满足 $e(c) \leq \varepsilon$, 则 $(x(c), y(c+1))$ 属于 R 集合, 转 Step 6;

Step 3 确定 w_c 的变化量 Δw_c , 修改 w_c ;

Step 4 对于集合 T , 更新各个样本系数 w_i , 依据 $e(i)$ 更新各个样本的分类 (E, S, R);

Step 5 判断 $(x(c), y(c+1))$ 是否满足 KKT 条件, 若满足, 则依据 $e(c)$ 确定属于 S 或者 E , 转 Step 6; 否则, 转 Step 3;

Step 6 得到 $(x(c), y(c+1))$ 的分类 (E, S, R), 算法结束.

在线支持向量回归算法的关键是修改 w_c 和 w_i ,

具体修改算法参考文献[6], 这里不再赘述.

4.2 自适应混沌优化学习算法 (Chaotic optimization learning algorithms)

混沌运动具有遍历性、随机性、“规律性”等特点, 混沌优化^[7] 是一种方便有效的直接优化算法. 考虑到优化效率和精度, 这里采用自适应混沌优化方法, 能够随着迭代次数的增加不断缩小搜索区间.

采用式 (13) 所示的 Logistic 混沌模型来产生混沌变量:

$$z_{n+1} = \mu z_n (1 - z_n), \quad n = 1, 2, \dots \quad (13)$$

将控制器的学习看作最小化控制误差, 以此得到混沌优化的目标函数:

$$\begin{cases} \min f(x_1, \dots, x_i) = \min J(W, K), \\ a_i \leq x_i \leq b_i. \end{cases} \quad (14)$$

这里 x_i 为优化变量, 包括网络权值 W 和比例因子参数 K , $[a_i, b_i]$ 为 x_i 的定义域. 取比例因子参数定义域为 $k_e \in [0.5, 150], k_{ec} \in [0.1, 10], k_u \in [0.01, 20]$. 自适应混沌优化算法描述如下:

Step 1 初始化. 给定最大迭代次数 N , 可以接受的最大误差 J_d , 用随机数产生 x_0 , 代入式 (13) 所示的混沌模型, 产生 i 个混沌变量 $x_{i, n+1} (i = 1, \dots, n)$, 取最优解 $x_i^* = x_i(0), f_i^* = f_i(0)$, 搜索范围取定义域 $a'_i = a_i, b'_i = b_i$, 置 $k = 1$;

Step 2 将每一个混沌变量线性映射到对应优化变量搜索区间:

$$x'_{i, n+1} = a'_i + (b'_i - a'_i)x_{i, n+1}; \quad (15)$$

Step 3 混沌迭代搜索. 用 $x_i(k)$ 表示 $x'_{i, n+1}$, 用 $f_i(k)$ 表示 $f(x_i(k))$, 则:

如果 $f_i(k) \leq f_i^*$, 则 $f_i^* = f_i(k), x_i^* = x_i(k)$;

如果 $f_i(k) > f_i^*$, 则 f_i^*, x_i^* 不变.

Step 4 若 $k > N$ 或 $f_i^* \leq J_d$ 时, 结束搜索; 否则继续进行搜索, 搜索区间进行自适应处理:

$$\begin{cases} a'_i = x_i^* - \phi(b_i - a_i), \\ b'_i = x_i^* + \phi(b_i - a_i). \end{cases} \quad (16)$$

ϕ 为收缩因子, 取 m 为较小的正整数, 定义 ϕ 为

$$\text{当 } k \geq m \text{ 时, } \phi = 1 - \left(\frac{k-m}{k}\right)^2;$$

$$\text{当 } k < m \text{ 时, } \phi = 1.$$

为了保证新搜索区间不越界, 作以下处理:

若 $a'_i < a_i$, 则 $a'_i = a_i$; 若 $b'_i > b_i$, 则 $b'_i = b_i$. 置 $k = k + 1$, 返回 Step 2 继续搜索.

自适应混沌优化与变尺度梯度优化是二种不同的学习算法, 不难发现, 混沌优化算法的计算更加简洁, 学习过程中不要求求导数, 因而可以不需要系统

辨识这个环节就可实现自学习. 变尺度梯度优化需要求导数, 而且需要系统辨识器, 计算比较繁杂, 控制性能受系统辨识精度的影响.

5 仿真研究 (Simulation study)

设被控对象为非线性系统 $y(k + 1) = \frac{y(k - 1) \cdot (0.1y(k) + 0.4)}{1 + y^2(k) + y^2(k - 1)} + 0.8y(k) + 0.8u(k) + 0.2u(k - 1) + v$, 其中 v 为噪声, $v(k + 1) = 0.5v(k) + 0.3v(k - 1) + 1.2(v(k) - 0.2)(v(k) - v(k - 1))$, 初始值 $y(0) = 0, y(1) = 0.01, u(0) = 0, v(0) = 0, v(1) = 0.1$. 取采样周期为 $0.01s$, 用设计的自学习控制器来跟踪正弦函数和方波函数, 这里将自学习控制器与模糊逻辑控制器进行了对比, 仿真图 3 和图 4 中, 自学习控制器 1 与自学习控制器 2 分别代表采用变尺度梯度优化与自适应混沌优化算法的自学习控制器.

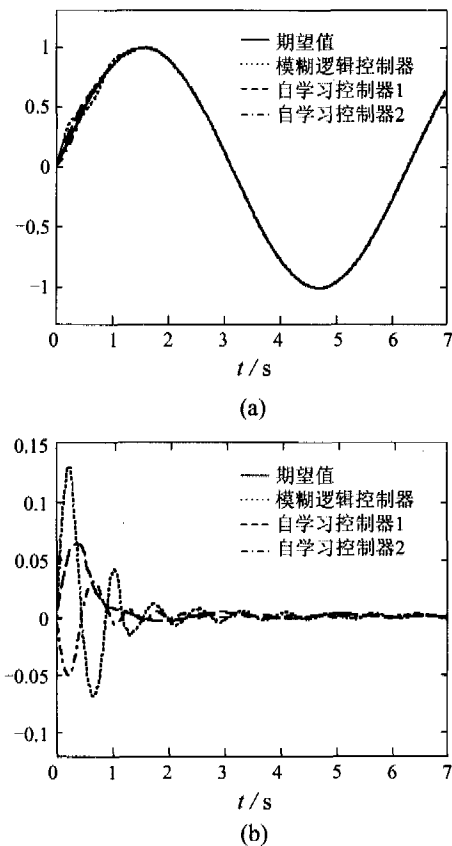


图 3 正弦函数跟踪控制仿真
Fig. 3 Simulation output results of sine wave

图 3 为正弦函数跟踪仿真结果, 图 3(a) 为实际跟踪效果, 图 3(b) 为跟踪误差曲线. 结果表明, 两种不同学习算法的自学习控制器跟踪控制性能都比较优, 控制误差比模糊逻辑控制器小的多, 受噪声和干扰的

影响小. 图 4 为方波函数跟踪仿真结果, 图 4(a) 为实际跟踪效果, 图 4(b) 为跟踪误差曲线. 结果表明, 自学习控制器响应速度较快, 没有超调量, 稳态精度更好, 而模糊逻辑控制器会产生控制超调量, 响应速度也慢些. 仿真研究表明, 两种不同学习算法的自学习控制器都具有比较优良的控制性能, 跟踪效果好, 在响应速度、控制精度等方面均优于模糊逻辑控制器, 在噪声和干扰存在时也能达到比较好的控制效果.

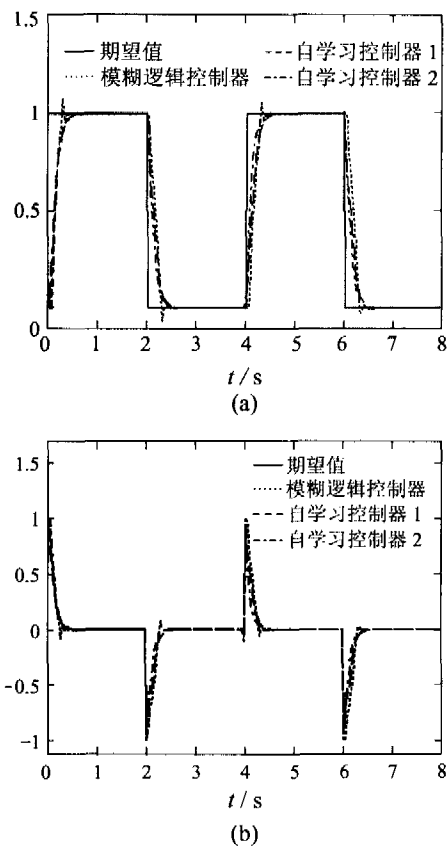


图 4 方波函数跟踪控制仿真
Fig. 4 Simulation output results of square wave

6 结论 (Conclusions)

文章利用支持向量机去实现模糊推理系统的自学习, 设计了支持向量机-模糊推理自学习控制器, 给出了控制器的结构和学习算法. 研究的两种学习算法中, 自适应混沌优化计算比较简洁, 不要求求导数, 可以不需要系统辨识环节; 变尺度梯度优化算法需要求导数, 计算比较复杂, 控制性能并受辨识器精度的影响. 自学习控制器既有模糊推理系统不依赖对象模型、鲁棒性强的特点, 又有支持向量机学习泛化能力强的特点, 针对非线性对象的仿真表明, 该控制器具有优良控制性能, 控制效果比模糊逻辑控制器更好.

参考文献(References):

- [1] LEE C C. Fuzzy logic in control systems; fuzzy logic controller-part 1 [J]. *IEEE Trans on Systems, Man, and Cybernetics*, 1990, 20(2): 404-418.
- [2] LIN C T, LEE C S G. Neural-network-based fuzzy logic control and decision system [J]. *IEEE Trans on Computers*, 1991, 40(12): 1320-1336.
- [3] VAPNIK V N. An overview of statistical learning theory [J]. *IEEE Trans on Neural Networks*, 1999, 10(5): 988-999.
- [4] 王定成, 方廷健. 一种基于支持向量机的内模控制方法[J]. *控制理论与应用*, 2004, 21(1): 85-88.
(WANG Dingcheng, FANG Tingjian. Internal model control approach based on support vector machines [J]. *Control Theory & Applications*, 2004, 21(1): 85-88.)
- [5] 王耀南. 智能控制系统(模糊逻辑、专家系统、神经网络控制) [M]. 长沙: 湖南大学出版社, 1996; 241-244.
(WANG Yaonan. *Intelligent Control Systems* [M]. Changsha: Hunan University Press, 1996; 241-244.)
- [6] JUNSHUI MA, JAMES THEILER, SIMON PERKINS. Accurate on-line support vector regression [J]. *Neural Computation*, 2003, 15(11): 2683-2703.
- [7] 李兵, 蒋慰孙. 混沌优化方法及其应用[J]. *控制理论与应用*, 1997, 14(4): 613-615.
(LI Bing, JIANG Weisun. Chaos optimization method and its application [J]. *Control Theory & Applications*, 1997, 14(4): 613-615.)

作者简介:

袁小芳 (1979—), 男, 博士研究生, 研究领域包括智能控制理论与应用、优化计算等, E-mail: yuanxiaof@21cn.com;

王耀南 (1957—), 男, 教授, 博士生导师, 研究领域包括智能控制理论与应用、智能信息处理、模式识别等, E-mail: yaonan@mail.hunu.edu.cn;

孙炜 (1975—), 男, 博士, 副教授, 主要研究领域包括机器人、智能控制理论与应用、神经网络等, E-mail: david-sun@tom.com.

下期要目

- 预测 PI 时滞网络拥塞控制算法设计及性能分析 钱艳平, 李奇, 刁翔
- 在逐点状态约束下一个四阶线性系统的时间最优控制 朱尚伟, **李训经**
- 一类 MIMO 非线性系统的稳定干扰解耦控制 宫清先, 张化光, 孟祥萍
- 基于启发式遗传算法的 SVM 模型自动选择 郑春红, 焦李成, 丁爱玲
- 不确定时滞分布参数系统鲁棒控制的 LMI 方法 罗毅平, 邓飞其
- 随机中立型泛函微分方程的 Lasalle 定理 沈轶, 江明辉, 廖晓昕
- 时变不确定系统的变时域离线鲁棒预测控制 丁宝苍, 邹涛, 李少远
- 求解最优潮流 KKT 系统的一类新模型及算法设计 罗可, 童小娇
- 基于多种群协同进化微粒群算法的径向基神经网络设计 王俊年, 申群太, 沈洪远, 周鲜成
- 变搜索区域多种群遗传算法 巩敦卫, 孙晓燕
- 粒子滤波进展与展望 杨小军, 潘泉, 王睿, 张洪才
- 随机马尔可夫切换系统的 H_2 模型降阶 孙敏慧, 邹云, 徐胜元
- 切换拟对称组合系统的降阶方法及稳定性分析 李建华, 李彦平, 赵军
- 两类模糊系统具有插值性的充要条件 侯健, 李洪兴, 王加银
- Hopfield 网络的全局指数稳定性 朱培勇, 孙世新
- 非线性采样控制系统的 Lagrange 稳定性 郭戈
- 两种最优观测融合方法的功能等价性 邓自立
- 具分段损失函数的支持向量机回归及在投资决策中的应用 胡根生, 邓飞其