

基于在线支持向量机和遗传算法的预测控制

陈进东, 潘 丰

(江南大学轻工过程先进控制教育部重点实验室, 江苏 无锡 214122)

摘要: 针对非线性系统模型预测控制中预测模型容易失配且目标函数难以求解的问题, 提出了一种基于在线支持向量机建模和遗传算法滚动优化的模型预测控制方法。该方法利用在线支持向量机建立被控对象的非线性模型, 在线支持向量机是一种迭代学习的支持向量机训练算法, 可以进行在线训练, 从而实现模型在线自校正; 并且通过遗传算法求解目标函数的最优控制量, 完成滚动优化。对非线性系统的仿真研究结果表明, 该方法有效且具有良好的自适应性。

关键词: 模型预测控制; 在线支持向量机; 遗传算法; 滚动优化

中图分类号: TP 273

文献标志码: A

DOI:10.3969/j.issn.1001-506X.2013.06.24

Online support vector machine and genetic algorithm based predictive control

CHEN Jin-dong, PAN Feng

(Key Laboratory of Advanced Process Control for Light Industry (Ministry of Education),
Jiangnan University, Wuxi 214122, China)

Abstract: Aiming at the problems that the predictive model is often mismatching and is difficult to solve the nonlinear optimization function of nonlinear system model predictive control, an online support vector machine (OSVM) modeling and genetic algorithm (GA) rolling optimization based model predictive control method is proposed. This proposed method builds a nonlinear model for objects using OSVM, which is an iterative support vector regression learning algorithm and can be used for online training, hence the predictive model parameters could be adjusted online through online learning. Furthermore, the nonlinear optimization function is solved by GA optimization, and rolling optimization is realized in the system. The nonlinear case simulation results show that the system adaptability is improved by the proposed method.

Keywords: model predictive control; online support vector machine (OSVM); genetic algorithm (GA); rolling optimization

0 引言

模型预测控制(model predictive control, MPC)是一种利用预测模型来控制被控对象输出的计算机控制算法^[1], 预测模型的准确性在 MPC 中具有很重要的作用。经典的预测控制算法, 一般是通过被控对象的单位阶跃响应或脉冲响应来估计所需模型参数, 得到其近似线性模型^[2]。但是对于强非线性被控对象, 这种方法就会出现模型失配的问题, 从而导致预测控制器性能下降。

针对传统 MPC 难以建立非线性对象的准确预测模型的缺点, 很多学者提出了基于智能模型的 MPC^[3]。智能模型无需对被控对象进行深入的了解, 只需对历史数据进行学习就可以建立比较准确的模型, 因此具有良好的应用前景。文献[4-6]分别设计了基于不同神经网络模型的预测

控制方法, 使之能够处理强非线性和带约束的对象, 但是神经网络具有容易陷入局部最优、泛化能力差、过训练等缺点; 因而文献[7-10]研究了基于支持向量机(support vector machine, SVM)的 MPC, SVM 具有全局最优、泛化能力强、小样本训练等特点, 有效提高了预测控制的效率。上述这些方法的预测模型都是离线训练模型, 不能在线对模型进行校正, 但非线性过程一般都具有时变性, 而离线建立的模型不能适应这种变化。在线支持向量机(online support vector machine, OSVM)是一种 SVM 在线训练方法^[11], 通过保存已有的训练参数, 当模型偏差超过设定值时, 通过在线学习调整模型参数, 使其重新满足设定精度, 因此 OSVM 能够适应非线性对象的动态变化。

MPC 最主要特征是滚动优化, 即通过使目标函数的最优来确定未来的控制输入, 并且在线反复进行优

化^[2]。由于预测模型越来越复杂,使得目标函数的直接最优求解变得十分困难,因此很多学者采用非线性优化方法对目标函数进行寻优。目前主要的方法有最速下降法、牛顿法、共轭梯度法等^[12],这些方法要求求解空间具有凸性,有的算法还要求函数具有二阶或高阶导数,因此不能作为通用的优化方法。仿生优化算法如遗传算法(genetic algorithm, GA)、粒子群优化(particle swarm optimization, PSO)算法等对所求解的问题的数学模型要求不高^[13],可以作为一种通用的优化方法。MPC一般为约束优化问题,而 PSO 算法在处理约束优化问题上,经常会出现不能收敛或结果为空集的情况。GA 是一种非线性全局优化算法,只需通过对初始种群的限制,就能很好地解决约束优化问题^[14],非常适合对有约束 MPC 的目标函数进行寻优求解。

因此,本文提出一种基于在线支持向量机和遗传算法的模型预测控制(online support vector machine and genetic algorithm based model predictive control, OSVM-GA-MPC)。利用 OSVM 的在线学习能力,在线自校正预测模型,从而适应非线性过程的动态变化;然后利用 GA 滚动优化目标函数,求得多步控制律,实现基于 GA 滚动优化的 MPC。

1 在线支持向量机原理

OSVM 是一种 SVM 在线训练方法,相对传统的 SVM 算法,OSVM 采用每次只增加一个新样本的方式,在线逐步迭代训练。

1.1 支持向量机

SVM 是从线性可分情况下的最优分类面发展而来,其基本思想可参见文献[15]。SVM 形式上类似一个神经网络,输出是中间节点的线性组合,每个中间节点对应一个支持向量,其结构如图 1 所示。

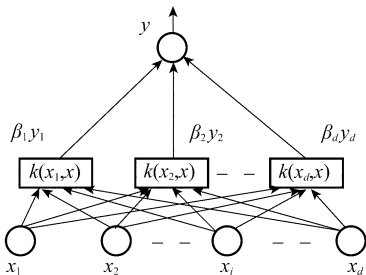


图 1 SVM 结构示意图

1.2 在线支持向量机

对于给定训练数据集 $T = \{x_i, y_i\} (i = 1, 2, \dots, l)$, $x_i \in \mathbf{R}^N$ 和 $y_i \in \mathbf{R}$, 非线性映射 Φ 将数据 x_i 映射到多维特征空间 F 。在此空间中,找到映射关系 f 能够很好地估计给定的数据,根据统计学习理论,构建回归方程为

$$f(x) = (\omega \cdot \Phi(x)) + b \tag{1}$$

$$\Phi: \mathbf{R}^N \rightarrow F$$

$$\omega \in F$$

式(1)所示函数估计问题等效为如下函数的最小化问题:

$$R_{\text{reg}}[f] = R_{\text{emp}}[f] + \lambda \|\omega\|^2 = \sum_{i=1}^l C(e_i) + \lambda \|\omega\|^2 \tag{2}$$

根据式(2)可以得到优化问题为

$$\min_{\omega, b} D = \frac{1}{2} \|\omega\|^2 + C \sum_{i=1}^l (\xi_i + \xi_i^*) \tag{3}$$

$$\text{s. t. } \begin{cases} y_i - \omega^T \cdot x_i - b \leq \epsilon + \xi_i \\ \omega^T \cdot x_i + b - y_i \leq \epsilon + \xi_i^* \\ \xi_i \geq 0 \\ \xi_i^* \geq 0 \end{cases}$$

求解式(3)可以求得 ω 和 b 。构建优化问题(3)的拉格朗日方程,此方程的对偶问题为

$$\min_{\alpha_i, \alpha_i^*, b} W = \frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l K(x_i, x_j) (\alpha_i - \alpha_i^*) (\alpha_j - \alpha_j^*) + \epsilon \sum_{i=1}^l (\alpha_i + \alpha_i^*) - \sum_{i=1}^l y_i (\alpha_i - \alpha_i^*) \tag{4}$$

$$\text{s. t. } \begin{cases} 0 \leq \alpha_i, \alpha_i^* \leq C_i, i = 1, 2, \dots, l \\ \sum_{i=1}^l (\alpha_i - \alpha_i^*) = 0 \end{cases}$$

式中, α_i 和 α_i^* 为拉格朗日乘子; $K(x_i, x_j) = \Phi(x_i)\Phi(x_j)$ 为核函数。

因此,可以构造一个回归方程为

$$f(x) = \sum_{i=1}^l (\alpha_i - \alpha_i^*) K(x_i, x) + b \tag{5}$$

通过拉格朗日乘子和 Karush-Kuhn-Tucker (KKT) 条件,训练数据可以被分为 3 个集合^[11],分别为误差支持向量集:

$$E = \{i = | |\theta_i| = C, |h(x_i)| \geq \epsilon\};$$

边界支持向量集:

$$S = \{i = | 0 < |\theta_i| < C, |h(x_i)| = \epsilon\};$$

剩余样本集:

$$R = \{i = | |\theta_i| = 0, |h(x_i)| \leq \epsilon\}.$$

其中, $\theta_i = \alpha_i - \alpha_i^*$ 和 $h(x_i) = f(x_i) - y_i$ 。

下面简单介绍 OSVM 算法的主要步骤,算法的具体实施可以参考文献[11]。

当新增一个样本 (x_c, y_c) 到训练集中,首先判断 $\Delta\theta_c$ 的改变是正还是负,即

$$\text{sign}(\Delta\theta_c) = \text{sign}(y_c - f(x_c)) = \text{sign}(-h(x_c)) \tag{6}$$

下一步为确定新增样本和原有样本的移动情况。为了简单起见,这里主要讨论 $\Delta\theta_c > 0$ 的情况, $\Delta\theta_c < 0$ 的情况与之类似。 $\Delta\theta_c$ 的取值大小为保证训练集中的样本只能移动到其邻近的训练集中,即不会出现从 R 集移动到 E 集或反之的情况。在 $\Delta\theta_c$ 确定后,根据 $\Delta\theta_c$ 调整 $h(x_c), \theta_c, h(x_i)$ 和 θ_i ,然后依照下面的规则移动样本。

对于一个新增样本 x_c ,

(1) 当 $h(x_c)$ 由 $h(x_c) < -\epsilon$ 变为 $h(x_c) = -\epsilon$ 时,将新

的样本 x_c 加入到 S 集中,并且结束此次迭代。

(2) 当 θ_c 由 $\theta_c < C$ 变为 $\theta_c = C$ 时,将新的样本 x_c 加入到 E 集中,并且结束此次迭代。

对于在 S 集中的每个样本 x_i ,

(3) 如果 θ_i 由 $0 < |\theta_i| < C$ 变为 $|\theta_i| = C$ 时,样本 x_i 由 S 集移动到 E 集;如果 θ_i 变为 $\theta_i = 0$ 时,样本 x_i 由 S 集移动到 R 集。

对于在 R 集中的每个样本 x_i ,

(4) 如果 $h(x_i)$ 由 $|h(x_i)| < \epsilon$ 变为 $|h(x_i)| = \epsilon$ 时,样本 x_i 从 R 集移动到 S 集。

对于在 E 集中的每个样本 x_i ,

(5) 如果 $h(x_i)$ 由 $|h(x_i)| > \epsilon$ 变为 $|h(x_i)| = \epsilon$ 时,样本 x_i 从 E 集移动到 S 集。

当 S 集发生变化时,需要对式(5)的参数重新调整。

2 算法原理和步骤

MPC 在非线控制中存在着诸多问题,其中主要的问题为难于建立有效的预测模型和求解目标函数;并且由于受环境影响,非线性对象容易发生偏移,而离线建立的预测模型无法适应这种变化。因此将 OSVM 与 MPC 相结合,并且利用 GA 滚动优化目标函数,从而实现自适应非线性 MPC。

OSVM-GA-MPC 的方框图如图 2 所示。图中 y_{sp} 表示参考轨迹 y_r 的设定值, y 表示实际的输出值, y_m 表示模型的预测输出, u 表示控制器输出, e 表示系统实际输出和模型预测输出之间的误差。

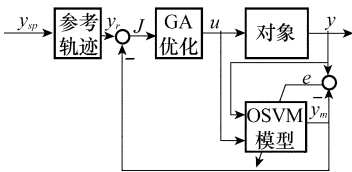


图 2 OSVM-GA-MPC 方框图

2.1 非线性预测模型

假定非线性对象为

$$y(k+1) = f[y(k), y(k-1), \dots, y(k-n_o+1), u(k), u(k-1), \dots, u(k-m_o+1)] \quad (7)$$

式中, f 为非线性函数;系统的阶次 n_o 和 m_o 都未知; $u_{\min} \leq u \leq u_{\max}$ 为控制器输出的限定范围。

OSVM 是基于核内积的非线性回归函数,模型的表达式为

$$y_m(k+1) = \hat{f}(\mathbf{I}_i) = \hat{f}[y(k), y(k-1), \dots, y(k-n'+1), u(k), u(k-1), \dots, u(k-m'+1)] \quad (8)$$

式中, \hat{f} 是基于 OSVM 的非线性函数; n' , m' 的阶次可以通过拟合的精确度来确定。

通过 l 组训练数据 $\{\mathbf{I}_i, y(i+1), i=1, 2, \dots, l\}$,可以得到对应的参数 $\theta_i (i=1, 2, \dots, l)$ 和偏离值 b 。所以在 k 时刻的下一步预测输出为

$$y_m(k+1) = \sum_{i=1}^l \theta_i (\mathbf{I}'_i \cdot \mathbf{I}_k + 1)^2 + b = \sum_{i=1}^l \theta_i [\mathbf{I}'_i(1)y(k) + \dots + \mathbf{I}'_i(n')y(k-n'+1) + \mathbf{I}'_i(n'+1)u(k) + \dots + \mathbf{I}'_i(n'+m')u(k-m'+1) + 1]^2 + b \quad (9)$$

为了获得多步预测输出,移动式(9)中的时域,并且假设 $n' \geq P \geq M, m' \geq P \geq M$, P 为预测时域, M 为控制时域。 $y_m(k+j)$ 为在 k 时刻的第 j 步预测输出, $P \geq j \geq 1$ 。当 $j \geq M$ 时,假设控制器的输出不变。

$$y_m(k+1) = \sum_{i=1}^l \theta_i [\mathbf{I}'_i(1)y(k) + \dots + \mathbf{I}'_i(n')y(k-n'+1) + \mathbf{I}'_i(n'+1)u(k) + \dots + \mathbf{I}'_i(n'+m')u(k-m'+1) + 1]^2 + b$$

$$y_m(k+2) = \sum_{i=1}^l \theta_i [\mathbf{I}'_i(1)y(k+1) + \dots + \mathbf{I}'_i(n')y(k-n'+2) + \mathbf{I}'_i(n'+1)u(k+1) + \dots + \mathbf{I}'_i(n'+m')u(k-m'+2) + 1]^2 + b$$

$$\vdots$$

$$y_m(k+M) = \sum_{i=1}^l \theta_i [\mathbf{I}'_i(1)y(k+M+1) + \dots + \mathbf{I}'_i(n')y(k-n'+M) + \mathbf{I}'_i(n'+1)u(k+M-1) + \dots + \mathbf{I}'_i(n'+m')u(k-m'+M) + 1]^2 + b$$

$$\vdots$$

$$y_m(k+P) = \sum_{i=1}^l \theta_i [\mathbf{I}'_i(1)y(k+P+1) + \dots + \mathbf{I}'_i(n')y(k-n'+P) + \mathbf{I}'_i(n'+1)u(k+P-1) + \dots + \mathbf{I}'_i(n'+m')u(k-m'+P) + 1]^2 + b \quad (10)$$

为了消除模型的失真和系统的干扰,引入模型参数在线校正。将预测误差 $e(k) = |y(k) - y_m(k)|$ 和设定误差 Err 比较,当 $e(k) < Err$,说明模型精度较高,不需要在线校正;当 $e(k) \geq Err$,则说明模型失配,需对其参数进行在线校正,在线校正的步骤可参见第 1.2 节中 OSVM 的训练算法。

2.2 滚动优化

OSVM-GA-MPC 的目标函数为

$$J(k) = \sum_{j=1}^P [y_r(k+j) - y_m(k+j)]^2 + \lambda \sum_{j=1}^M [\Delta u(k+j-1)]^2 \quad (11)$$

式中, λ 为控制的权系数; $u(k)$ 为控制信号; Δ 为差分运算符, $\Delta u(k) = u(k) - u(k-1)$; $y_m(k+j)$ 是 OSVM 的第 j 步预测值, $P \geq j \geq 1$, $y_r(k+j)$ 是第 j 步参考输出,表示为

$$\begin{cases} y_r(k) = y(k) \\ y_r(k+j) = \alpha_r y_r(k+j-1) + (1-\alpha_r) y_{sp}(k) \end{cases} \quad (12)$$

式中, α_r 是控制系统的鲁棒性和收敛性相关的调整因子,

$0 < \alpha_r < 1; y_{sp}(k)$ 为真实的设定值。

采用 GA 进行控制量的寻优,可以分为 4 个部分:遗传算法编码,生成初始种群及约束引入,选择、交叉和变异操作,终止判定。

2.2.1 遗传算法编码

采用二进制编码,完成变量基因型和表现型的映射关系。将步长为 M 的一组控制变量表示为一个个体,单个控制量用长度为 $L_u = \left\lfloor \log_2 \frac{(u_{\max} - u_{\min})}{scale} \right\rfloor$ 的二进制串表示($\lfloor x \rfloor$ 表示大于 x 的最小整数, $scale$ 为控制量的精度),编码二进制串的总长度为 $L = L_u \times M$ 。

2.2.2 生成初始种群及约束引入

在控制开始的第一时刻,则在初始种群的产生过程中通过空间限定法引入约束。由于式(11)中的约束采用的上下限形式,只需在区间 (u_{\min}, u_{\max}) 中随机选取 $n = \delta(M + L_u)$ 个个体表示初始种群, δ 一般取 8, 这样便自动将约束加入到了初始种群中。如果是 $k + 1$ 时刻,则将 k 时刻优化结果中的后 $M - 1$ 个基因左移一位,最后一个基因保持不变,构成 $k + 1$ 时刻的一个初始个体,如图 3 所示。在初始种群中以一定数量比例 α 放入该个体的同时,为保证初始种群的多样性,其余 $1 - \alpha$ 比例的个体随机产生,一般 $\alpha = 0.01$ 。

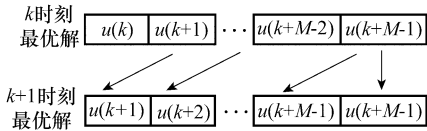


图 3 $k + 1$ 时刻初始总群的产生

2.2.3 选择、交叉和变异操作

由于遗传算法的目标函数是求最大值,因而不能直接将式(11)直接用作遗传算法优化的适应值函数,将其修改为

$$F(k) = \frac{1}{1 + J(k)} \quad (13)$$

通过该形式的变化,便将模型预测控制优化的极小值问题转化为遗传算法优化的极大值问题。

用排序选择法进行选择操作,即对群体中的所有个体按其适应值大小进行降序排列,然后进行比例选择产生下一代种群;用算术交叉法进行交叉操作;按概率随机在 (u_{\min}, u_{\max}) 中选取 1 个个体代替原来的个体进行变异操作。

2.2.4 终止判定

采用两种方法判定结束:

(1) 通过适应值函数的值判断。如果遗传过程中出现适应度函数值的为 1 的个体,则它为最优个体。

(2) 通过最优个体适应度函数的收敛程度判断。给定一个判断算法收敛程度的代数 c ,如果在连续 c 代中,每代种群中的最优个体都没有发生变化,则终止计算,选择该个体为最优个体。

当上述两个条件有一个满足时,便可以停止计算,选取最优个体对应的控制量作为当前最优解,然后输出。

2.3 算法步骤

OSVM-GA-MPC 的整体步骤如下:

步骤 1 选择合适的参数,包括 $C, \epsilon, Err, n', m', \alpha_r, P, M, \lambda$ 和 η_r 。关于 C 和 ϵ 的选取在文献[15]中做了讨论, Err 由文献[16]参考选取,参数 n' 和 m' 的选取和神经网络情况相似,可以通过文献[17]所定的规则选取,其他参数 $\alpha_r, P, M, \lambda, \eta_r$ 可以参考文献[1]。

步骤 2 为了提高模型的运算速率,先对模型进行离线训练,得到 $\theta_i (i = 1, 2, \dots, l)$ 和 b 。

步骤 3 在采样时刻 k ,通过比较模型输出和系统实际输出得到其预测误差 $e(k)$ 。

步骤 4 如果 $e(k) < Err$,则将此输入输出向量加入训练集,OSVM 模型保持不变;如果 $e(k) \geq Err$,则先将此输入输出向量加入训练集,然后在已有的模型参数基础上,通过在线训练方法,逐步调整预测模型参数。

步骤 5 通过式(12)计算参考估计 $y_r(k + j) (j = 1, 2, \dots, P)$ 。

步骤 6 通过 GA 优化得到最优控制序列 $u(k + j - 1) (j = 1, 2, \dots, M - 1)$,当 $j \geq M$ 时, $u(k + j - 1) = u(k + M - 1)$ 。

步骤 7 通过式(10)和已知的 $u(k + j - 1)$,从而得到 $y_m(k + j | k) (j = 1, 2, \dots, P)$ 。

步骤 8 将第一个控制器输出 $u(k)$ 应用到对象,得到系统的输出 $y(k + 1)$ 。

步骤 9 增加 k ,即 $k + 1 \rightarrow k$,回到步骤 3,重复整个运算过程。

3 仿真研究

在连续搅拌反应釜(continuous stirred tank reactor, CSTR)中发生的简单一级不可逆放热反应,采用文献[18]所研究的模型,其物质和能量平衡的模型方程为

$$x_1(t) = -\frac{1}{\lambda}x_1(t) + D_a[1 - x_1(t)] \cdot$$

$$\exp\left[\frac{x_2(t)}{1 + x_2(t)/\gamma_0}\right] + \left(\frac{1}{\lambda} - 1\right)x_1(t - \tau) \quad (14)$$

$$x_2(t) = -\left(\frac{1}{\lambda} + \beta\right)x_2(t) + HD_a[1 - x_1(t)] \cdot$$

$$\exp\left[\frac{x_2(t)}{1 + x_2(t)/\gamma_0}\right] + \left(\frac{1}{\lambda} - 1\right)x_2(t - \tau) + \beta u(t) \quad (15)$$

式中, $\gamma_0 = 20; H = 8; \beta = 0.3; D_a = 0.072; \lambda = 0.8; \tau = 2$ 。

状态 $x_1(t)$ 是反应的转换速度,且 $0 < x_1(t) < 1; x_2(t)$ 是温度,但无量纲。假设只有温度能够在线测量,且

$$y(t) = [0 \quad 1] \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} \quad (16)$$

针对以上模型,采用本文所提出的方法进行建模与控制。在建模的时候输入采用 $[-1, 1]$ 之间的随机数,共产生 4 批数据,3 批训练,1 批测试。采用 OSVM 进行离线建模时,OSVM 的参数通过交叉验证法确定,得到 $n' = 5, m' = 5, C = 300$ 和 $\epsilon = 0.1$,模型的预测误差如图 4 所示。

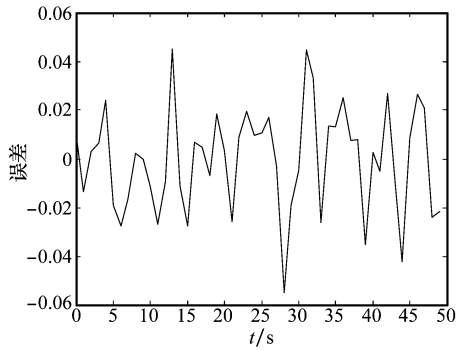


图4 OSVM对于CSTR建立预测模型的误差

从图4可以看出,通过OSVM对CSTR的反应过程建立的预测模型,模型的预测误差较小,基本不超过0.03,设定 $Err=0.03$ 。

在系统进入稳态之后,给定的信号为阶跃信号。设定优化时域 $P=5$,控制时域 $M=3$,柔化系数 $\alpha_r=0.2$, $\lambda=0.05$, $\gamma_k=0.02$;由建模可知 $u_{\min}=-1$, $u_{\max}=1$ 。对于GA参数设定:控制的精度 $scale=0.001$,则 $L_c=11$;设定连续的代数 $c=10$ 。

为了更好地说明本方法的有效性,将OSVM-GA-MPC与文献[8]所述的基于支持向量机的模型预测控制(support vector machine based model predictive control, SVM-MPC)进行比较,结果如图5所示。

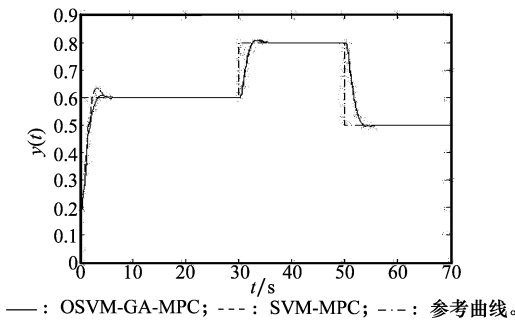


图5 OSVM-GA-MPC对CSTR仿真结果

由图5可以看出,在模型匹配的情况下,OSVM-GA-MPC和SVM-MPC的控制结果相似,都能在较短的时间内,将系统输出稳定在设定值上,并且没有稳态误差。

将CSTR模型的参数 $\gamma_0=20$ 和 $\lambda=0.8$ 加以调整,分别增大5%,出现模型失配的情况时,OSVM-GA-MPC和SVM-MPC的控制效果出现较大差异,结果如图6所示。

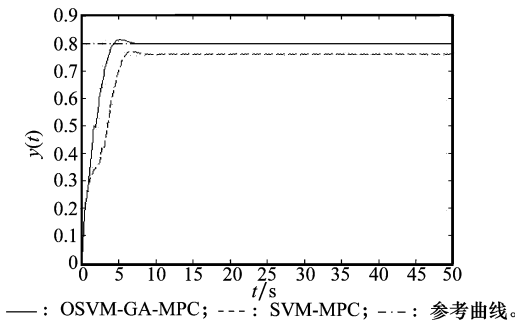


图6 OSVM-GA-MPC对CSTR模型失配的控制效果

从图6可以看出,由于具有在线学习能力,当模型出现偏移时,OSVM-GA-MPC仍然能够很快地将输出稳定在设定值上,超调很小,没有稳态误差。但是SVM-MPC由于不具有在线学习能力,当模型出现偏移时,虽然也能将系统输出稳定,但是存在较大的稳态误差。

4 结论

本文针对强非线性对象提出了一种OSVM-GA-MPC方法。先通过OSVM建立非线性对象的预测模型,并利用OSVM的在线学习能力,实现了模型在线自校正;再通过GA对目标函数求解,实现了自适应控制。通过非线性对象的仿真研究,验证了该方法不仅在模型精确的情况下,控制结果较好;即使在模型失配的情况下,也能通过在线学习校正预测模型,从而消除模型失配带来的影响,取得较好的控制效果,因此具有很好的自适应性能。

参考文献:

- [1] Shu D Q. *Predictive control system and its application* [M]. Beijing: China Machine Press, 1996: 1-36. (舒迪前. 预测控制系统及其应用[M]. 北京: 机械工业出版社, 1996: 1-36.)
- [2] Xi Y G. *Predictive control* [M]. Beijing: National Defense Industry Press, 1993: 5-18. (席裕庚. 预测控制[M]. 北京: 国防工业出版社, 1993: 5-18.)
- [3] Cheng H, Liu Z Y. Nonlinear model predictive control: the state and open problems[J]. *Control and Decision*, 2001, 16(4): 385-391. (陈虹, 刘志远. 非线性模型预测控制的现状与问题[J]. 控制与决策, 2001, 16(4): 385-391.)
- [4] Hosen M A, Hussain M A, Mjalli F S. Control of polystyrene batch reactors using neural network based model predictive control (NNMPC): an experimental investigation[J]. *Control Engineering Practice*, 2011, 19(5): 454-467.
- [5] Han H G, Qiao J F, Chen Q L. Model predictive control of dissolved oxygen concentration based on a self-organizing RBF neural network[J]. *Control Engineering Practice*, 2012, 20(4): 465-476.
- [6] Dalamagkidis K, Valavanis K P, Piegel L A. Nonlinear model predictive control with neural network optimization for autonomous autorotation of small unmanned helicopters[J]. *IEEE Trans. on Control Systems Technology*, 2011, 19(4): 818-831.
- [7] Shin J, Jin K H, Kim Y. Adaptive support vector regression for UAV flight control[J]. *Neural Networks*, 2011, 24(1): 109-120.
- [8] Zhong W, Pi D. Support vector machine based nonlinear model multi-step-ahead optimizing predictive control[J]. *Journal of Central South University of Technology*, 2005, 12(5): 591-595.
- [9] Zhang R D, Wang S Q, Li P. Support vector machine based pre-

- dictive control for nonlinear systems[J]. *Acta Automatica Sinica*, 2007, 33(10):1066-1073. (张日东, 王树青, 李平. 基于支持向量机的非线性系统预测控制[J]. 自动化学报, 2007, 33(10): 1066-1073.)
- [10] Mu Z X, Zhang R M, Sun C Y. LS-SVM predictive control based on PSO for nonlinear systems[J]. *Control Theory and Application*, 2010, 27(2):164-168. (穆朝絮, 张瑞民, 孙长银. 基于粒子群优化的非线性系统最小二乘支持向量机预测控制方法[J]. 控制理论与应用, 2010, 27(2):164-168.)
- [11] Ma J, Theiler J, Perkins S. Accurate on-line support vector regression[J]. *Neural Computation*, 2003, 15(11): 2683-2704.
- [12] Qin S J, Badgwell T A. A survey of industrial model predictive control technology[J]. *Control Engineering Practice*, 2003, 11(7):733-764.
- [13] Verma H, Jain C, Rathore A, et al. A comparative study of GA, PSO and Big Bang-Big crunch optimization techniques for optimal placement of SVC's[J]. *International Journal of Electronics Communication and Computer Engineering*, 2012, 3(3): 263-269.
- [14] Goldberg D E. *Genetic algorithms in search, optimization, and machine learning*[M]. Boston: Addison-Wesley Longman Press, 1989: 30-35.
- [15] Vapnik V. *The nature of statistical learning theory*[M]. New York: Springer, 1999: 3-30.
- [16] Wang H, Pi D Y, Sun Y X. Online SVM regression algorithm-based adaptive inverse control[J]. *Neurocomputing*, 2007, 70(4): 952-959.
- [17] Noriega J R, Wang H. A direct adaptive neural-network control for unknown nonlinear systems and its application[J]. *IEEE Trans. on Neural Networks*, 1998, 9(1): 27-34.
- [18] Bao Z J, Pi D Y, Sun Y X. Nonlinear model predictive control based on support vector machine with multi-kernel[J]. *Chinese Journal of Chemical Engineering*, 2007, 15(5): 691-697.

作者简介:

陈进东(1983-),男,博士研究生,主要研究方向为工业过程建模与控制。

E-mail:chenjd83@gmail.com

潘 丰(1963-),男,教授,博士,主要研究方向为工业过程建模与控制。

E-mail:pan_feng_63@163.com