

一种双故障情形的主动滑模容错控制方法

朱大奇, 李晓妮

(上海海事大学 信息工程学院, 上海 200135)

摘要:针对非线性系统发生多故障情形的主动容错控制,利用平衡学习的CMAC神经网络作为故障诊断的手段,在线估计故障函数,用滑模控制算法进行容错控制律的在线重构,实现非线性系统主动容错控制。以一双故障模式为例进行仿真,结果证明所提方法有效。

关键词:容错控制;非线性系统;滑模控制;故障诊断

中图分类号: TP273 **文献标识码:** A **文章编号:** 1009-3516(2007)04-0015-04

主动容错控制器设计的一个重要方法是基于FDD(Fault Detection and Diagnosis)的控制律在线重构策略。它根据系统在线实时故障诊断结果,通过一定算法对控制系统的控制规律进行在线实时重构,是容错控制领域中倍受关注的研究方向之一。目前在此方面虽然已有一些研究报道,但大多数是针对单故障情况的故障诊断与容错控制研究^[1-4],然而实际中常有可能多个故障同时发生,因此针对多故障的容错控制是很有实际意义的。本文引入文献[5]给出的基于平衡学习的CMAC(Cerebellar Model Articulation Controller)神经网络非线性辨识算法,进行在线故障辨识,有效提高故障诊断的实时性和精确性。在此基础上结合文献[6]给出的变结构滑模控制算法重构故障系统的控制规律,给出一种多故障情形下的非线性系统主动容错控制方法。

1 基于平衡学习的CMAC神经网络在线故障评估器

对主动容错控制来说,首要的任务是在线辨识故障,本文利用文献[5]提出的基于平衡算法的小脑神经网络作为故障在线辨识的方法。该方法是在基于信度分配的CMAC神经网络模型基础上的改进算法,其算法实现过程仍包含两个阶段:CMAC结果输出阶段和权值学习阶段。在结果输出阶段,CMAC将所有被输入状态激活的存储单元的权值(存储数据)相加,即

$$y_i = \sum_{j=1}^{N_L} w_j a_j(x) \quad i = 1, 2, \dots, m \quad (1)$$

式中: y_i 为状态 i 的实际输出; m 为总状态数; w_j^{n-1} 为第 j 个存储单元的权值,若 $a_j(x)$ 激活,则其值为1,否则为0,只有 N_L 个存储单元对输出有影响。在权值学习阶段,其权值调整公式为:

$$w_j^k = w_j^{k-1} + \beta_1 \left\{ \frac{(f(j) + 1)^{-\mu}}{\sum_{j=1}^{N_L} (f(j) + 1)^{-\mu}} \right\} (y_d - y_i) \quad (2)$$

式中: y_d 是CMAC期望输出; μ 为平衡学习常数。当 μ 为0或1时,ICA-CMAC分别为常规CMAC^[7]和基于信度分配的CMAC^[8](CA-CMAC)。平衡学习常数 μ 反映了网络训练过程中,先前学习的信息与未学习或少学习信息对存储单元权值调整的影响程度。不同的 μ 会有不同的学习结果,当 μ 为某一中间数值时,其学习速度最快,说明此时网络的“记忆”与“遗忘”达到了最佳平衡。在容错控制中,可以用此算法有效提高故障诊断的实时性,更具体的算法实现见文献[5]。

收稿日期:2007-03-29

基金项目:国家“863”高新技术研究与发展计划项目(2006AA09Z210)和上海市教委科研项目资助(06FZ038)

作者简介:朱大奇(1964-),男,安徽安庆人,教授,博士生导师,主要从事故障诊断与容错控制研究。

2 基于离散滑模控制的非线性系统控制律重构

考虑如下未知的 SISO 非线性故障系统:

$$y(k+1) = y(k) + \Delta t \hat{f}(y(k)) + \Delta t g(y(k))u(k) + \sum_{i=1}^n \beta_i(k - T_i) f_i(y(k), u(k)) \quad (3)$$

式中: $g(y(k)) = \hat{g}(y(k)) + \Delta g(\cdot)$; $\hat{f}(y(k))$ 和 $\hat{g}(y(k))$ 为状态矢量 $y(k)$ 的已知非线性函数; Δg 为有界不确定项; Δt 为采样周期; $\beta_i(k - T_i)$ 为故障类型; T_i 是故障发生的时间; $f_i(\cdot)$ 代表第 i 个未知故障模式, 设为系统输出矢量的函数; n 为系统可能发生的故障总数。

定义 $F(k)$ 为系统第 k 步的故障数值, 一般说来, $F(k)$ 是未知的。可用在线故障评估器(用基于平衡算法的 CMAC 神经网络: ICA - CMAC) 来近似它, $F(k)$ 可表达为 $F(k) = NF(k) + \Delta NF(k)$, $NF(k)$ 为在线故障评估器的输出, $\Delta NF(k)$ 为在线故障评估器的故障辨识误差, 可以看作系统的不确定性, 利用文献 [1, 6, 9] 的离散非线性不确定系统滑模控制规律的设计思想(具体推导见文献 [1, 6])。在动态系统故障模式下, 容错控制律 $u(k)$ 可以在正常控制律 $u_1(k)$ 的基础上增加一纠错控制律 $u_2(k)$ 来实现, 这样容错控制律 $u(k) = u_1(k) + u_2(k)$ 。其中正常控制律 $u_1(k)$ 和纠错控制律 $u_2(k)$ 分别为

$$u_1(k) = \frac{D(k)}{D(k)\hat{g}(y(k))} \left[-\hat{f}(y(k)) + \frac{\Delta y_d}{\Delta t} \right] = \frac{1}{\hat{g}(y(k))} \left[-\hat{f}(y(k)) + \frac{\Delta y_d}{\Delta t} \right] \quad (4)$$

$$u_2(k) = \frac{K(k)}{D(k)g(\cdot)} \text{sat} \left(\frac{S(K)}{\phi(k)} \right) + U(k - T_c) \frac{-NF(\cdot)}{g(\cdot)} \quad (5)$$

式(5)第一项与神经网络故障评估器的故障辨识误差有关。其第二项 $U(k - T_c) \frac{-NF(\cdot)}{g(\cdot)}$ 是用来补偿正常

控制器的纠正信号。 $NF(\cdot)$ 是跟踪故障 $\sum_{i=1}^n \beta_i(\cdot) f_i(\cdot)$ 的在线评估器输出。 T_c 为一用来评估控制效果的特别时间段。 $\phi(k)$ 为边界层厚度, $K(k)$ 是控制器增益, 饱和函数定义为

$$\text{sat} \left(\frac{S(K)}{\phi(k)} \right) = \begin{cases} +1 & \text{if } S(k) > \phi(k) \\ \frac{S(K)}{\phi(k)} & \text{if } |S(k)| \leq \phi(k) \\ -1 & \text{if } S(k) < -\phi(k) \end{cases} \quad (6)$$

$$\phi(k) = \eta(k) + \varepsilon, \quad K(k) = \eta(k) + 2\varepsilon \quad (7)$$

$$\eta_{\text{new}}(k) = \begin{cases} \sup_L \{ |D(k) \Delta NF(\cdot)| \} = \sup_L \{ |D(k) (\sum_{i=1}^n \beta_i(\cdot) f_i(\cdot) - NF(k))| \} & \Omega \leq \delta \\ \eta_{\text{old}} & \text{otherwise} \end{cases} \quad (8)$$

$$S(k) = \frac{y_d(k) - y_d(k-1)}{\Delta t} - \frac{y(k) - y(k-1)}{\Delta t} + a(y_d(k) - y(k)), \quad D(k) \approx \frac{S(k) - S(k-1)}{y(k) - y(k-1)},$$

$$\tilde{y}(k) = y_d(k) - y(k) \quad (9)$$

式(7)中 ε 是一微小的正数, $\eta(k)$ 则按式(8)动态更新。式(9)中 $y_d(k)$ 和 $y(k)$ 分别是系统第 k 步的期望输出和实际输出。 $a \in R^+$ 定义为系统输出收敛到期望输出的速度。 $\phi(k)$ 是边界层厚度, 故障辨识误差 ΔF

$= \Delta NF(k) = \sum_{i=1}^n \beta_i(\cdot) f_i(\cdot) - NF(k)$ 为系统的故障不确定性。参数 l 表示一时间周期, 在每一时间段 $L = [k - l, k]$, 故障辨识误差的上界被评估。式(7)的边界层厚度和控制器增益根据在线评估器输出在线自动调整, 以减少控制误差。在线学习结果按下列准则进行评估:

$$\text{SSAE } 0 = \sum_{k=k_0}^{k_0+l-1} (fy(k) - nfy(k))^2, \quad \text{SSAE } 1 = \sum_{k=k_0+l}^{k_0+2l-1} (fy(k) - nfy(k))^2 \quad (10)$$

$$\Omega = |\text{SSAE } 1 - \text{SSAE } 0| \quad (11)$$

此处 SSAE 0 和 SSAE 1 代表两个连续窗口在线评估器近似误差的平方和。 $nfy(k)$ 是评估器输出, $fy(k)$ 为系统第 k 步实际测试值与正常模型输出的差值。 Ω 为在线评估器参数, δ 为一阈值。当 $\Omega \leq \delta$ 时, 表明在线评估器足够精确, ICA - CMAC 网络收敛。

3 容错控制系统仿真

3.1 非线性模型

取下列一 SISO 的正常模型为

$$y(k+1) = y(k) - \frac{\Delta t}{m}(cy(k)^3 + k_1y(k-1)) + \frac{\Delta t}{m}u(k) \quad (12)$$

式中: $y(k+1)$ 和 $u(k)$ 代表系统第 $(k+1)$ 步时的输出和第 k 步的控制输入; Δt 是采样时间; c, m, k_1 是系统参数, 假定已知。为了跟踪期望曲线 $y_d(k+1)$ 则正常情况的控制输入 $u_1(k)$ 由式(12) 决定:

$$u_1(k) = \frac{m}{\Delta t}(-y(k) \frac{\Delta t}{m}(cy(k)^3 + k_1y(k-1))) + y_d(k+1)) \quad (13)$$

当系统发生未预料的故障时, 系统模型为

$$y(k+1) = y(k) - \frac{\Delta t}{m}(cy(k)^3 + k_1y(k-1)) + \frac{\Delta t}{m}u(k) + \sum_{i=1}^n \beta_i(k-T_i)f_i(y(k), y(k-1), u(k)) \quad (14)$$

本文以先渐进后突发的双故障情况为例进行仿真研究。渐进性故障可表示为: $\beta(k-T) = (1 - e^{-\alpha(k-T)})U(k-T)$, 突发性故障可表示为: $\beta(k-T) = U(k-T)$, 采样时间 $\Delta t = 0.01$, 系统参数 $c = 5, k = 100, m = 1$ 神经网络学习率 $\beta_1 = 1$, 平衡学习参数 $\mu = 1.2$ 。参考输入: $r(k) = 0.2\sin(k\pi/20)$, 期望输出: $y_d(k+1) = 0.6y_d(k) + 0.2y_d(k-1) + r(k)$, 其他仿真参数为 $\omega = 5, \lambda = 10^{-5}, \eta_0 = 0.5, \delta = 0.001, l = 10$ 。

3.2 仿真结果分析

$$\beta_1(k-T_1) = (1 - e^{-0.02(k-T_1)})U(k-T_1), \beta_2(k-T_2) = U(k-T_2), T_1 = 50, T_2 = 100 \quad (15)$$

$$f_1(y(k), y(k-1)) = 1 - e^{-0.7|y(k)-y(k-1)|}, f_2(y(k), u(k)) = 0.46y(k)y(k-1) \quad (16)$$

仿真结果见图 1-图 4。当仅有正常控制器单独作用时, 图 1 显示了系统实际输出与期望输出曲线, 在 50 周期时发生第一个故障, 系统并没有检测到此故障, 仍然按正常控制律运行到 100 周期, 当第二个突发故障发生时, 系统的输出开始出现越来越大的偏移, 并且在运行到 225 周期时失去控制。图 2 显示加入纠错控制律后系统运行情况, 很明显系统控制性能得到极大改善, 这是由于在 100 个周期时故障被系统检测出来, 在此时刻 ICA-CMAC 神经网络故障估计器被初始化, 并在线学习两个未知故障的综合作用结果, 同时加入补偿控制信号。系统在经过小幅振荡后重新稳定运行。图 3 显示了神经网络的辨识过程, 在 100 周期之前并无辨识误差, 这是因为在第 100 个周期之前, 系统并未发现故障; 在此之后才对故障估计器进行初始化, 故障估计器开始学习故障, 随着学习时间的延长, 神经网络输出逐渐收敛, 辨识误差逐渐减小。图 4 可以看出实际的控制输入在 100 周期之前只有正常控制器作用于系统中, 在 100 到 125 周期之间它是一个变化的数值, 加入容错控制律时, 故障估计器还没收敛, 所以变化较大。125 周期后控制信号稳定, 系统也稳定。

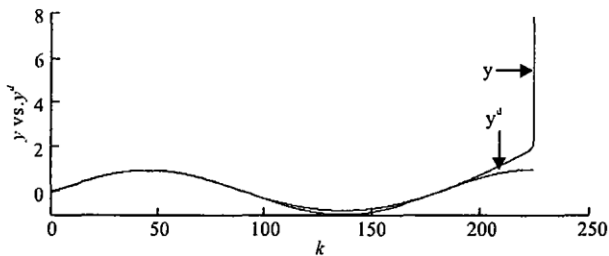


图 1 正常控制器单独作用时系统实际输出与期望输出

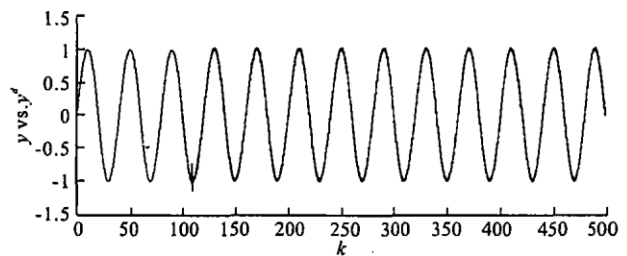


图 2 增加容错控制律后系统的实际输出与期望输出

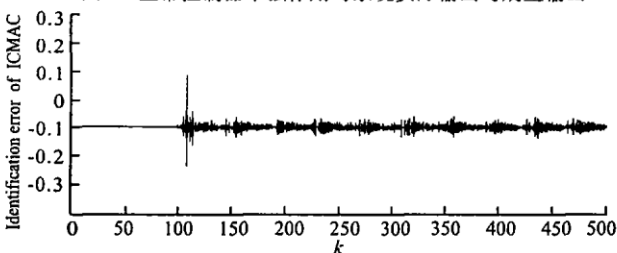


图 3 CA-CMAC 在线故障估计器的辨识误差

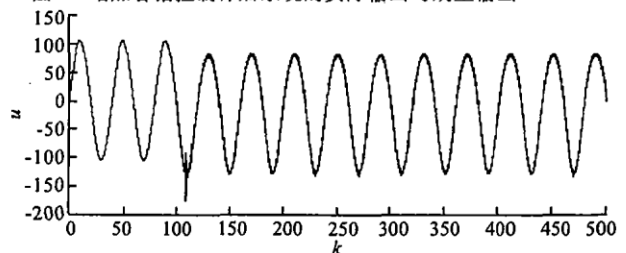


图 4 系统实际的控制信号

4 结论

针对非线性系统多故障情况,提出了运用基于平衡算法的小脑神经网络实现故障模式的在线学习,并利用滑模变结构控制进行控制律重构维持系统的正常运行,通过先渐进后突发性故障模式的仿真分析,表明了该容错控制方法能够成功驱动故障系统回到期望曲线,具有较好的控制效果。

参考文献:

- [1] Zhu Da - qi, Min Kong Fault - Tolerant Control of Dynamic Nonlinear System Using Credit assign fuzzy CMAC[J]. ACTA Automatica Sinica, 2006,32(3):329 - 336.
- [2] Polycarpou M M. Fault Accommodation of a Class of Multivariable Nonlinear Dynamical Systems using a Learning Approach[J]. IEEE Transactions on Automatic Control, 2001,46(5):737 - 742.
- [3] Zhang X. Parisini T. Polycarpou. M. M, Adaptive Fault - Tolerant Control of Nonlinear Uncertain Systems: An information - Based Diagnosis Approach[J]. IEEE Transactions on Automatic Control, 2004,49(8):1259 - 1274.
- [4] Ashari A E. Sedigh A K. Yazdanpanah M J. Reconfigurable Control System Design Using Eigenstructure assignment: Static, Dynamic and Robust Approaches[J]. International Journal of Control, 2005,78(13):1005 - 1016.
- [5] 朱大奇,张 伟. 基于平衡学习的 CMAC 神经网络非线性辨识算法[J]. 控制与决策,2004,19(12):1425 - 1429.
- [6] Misawa E. Discrete - Time Sliding Mode Control for Nonlinear Systems With Unmached Uncertainties and Uncertain Control Vector[J]. ASME Journal of Dynamic Systems, Measurement, and control, 2000,119(9):4204 - 4208.
- [7] Albus J S. Data Storage in Cerebellar Model Articulation Controller(CMAC)[J]. ASME J. Dynamic Systems, Measurement, Control, 1975,97(2):228 - 233.
- [8] Shun - Feng S, Ted T, Hung T H. Credit Assigned CMAC and Its Application to Online Learning Robust Controllers[J]. IEEE Trans. On Systems, Man, and Cybernetics - partB: Cybernetics, 2003,33(2):202 - 213.
- [9] Nicolas Roddier. Atmospheric Wavefront Simulation Using Zernike Polynomials[J]. Optical Engineering, 1990,29(10):1174 - 1180.

(编辑:姚树峰)

A Method of Active Fault - tolerant Control Method Based on Sliding Model Control for Double - fault Situation

ZHU Da - qi, LI Xiao - wei

(Information Engineering College, Shanghai Maritime University, Shanghai 200135, China)

Abstract: For the active fault - tolerant control method of non - linear system with multi - failures, the identification algorithm of the improved CMAC based on balanced learning is used to estimate faults function in this paper. The fault - tolerant control is realized by utilizing the effective control law reconfiguration strategy based on sliding model control algorithm. Finally, a simulation is done for the examples of double - fault cases. The simulation results demonstrate the effectiveness of the proposed tolerant - fault control method.

Key words: fault - tolerant control; non - linear system; sliding model control; fault diagnosis