

一类不确定离散混沌系统的自抗扰控制器与小脑神经网络并行优化控制*

刘朝华^{12))†} 章 兢¹⁾ 张英杰²⁾ 李小花³⁾ 吴建辉²⁾

1) (湖南大学电气与信息工程学院, 长沙 410082)

2) (湖南大学计算机与通信学院, 长沙 410082)

3) (长沙师范学校, 电子信息工程系, 长沙 410100)

(2010年5月30日收到; 2010年7月3日收到修改稿)

针对一类受扰不确定离散非线性混沌系统, 提出了基于免疫动态微粒群优化策略的ADRC与CMAC神经网络并行控制方法(ADRC-CMAC). ADRC控制器抑制系统扰动, 保证系统的稳定性; CMAC神经网络控制器实现前馈控制保证系统的控制精度和响应速度. 利用动态免疫微粒群算法对ADRC-CMAC并行控制器参数进行全局优化. 实验结果表明该控制方法具有较快系统的响应速度, 较好的抗干扰能力, 控制精度高.

关键词: 自抗扰控制器, 小脑神经网络, 并行控制, 混沌系统

PACS: 07.05.Mh, 95.10.Fh, 02.30.Yy, 05.45.Gy

1. 引言

离散非线性混沌系统在数据加密^[1-3]、保密通信^[4-6]、混沌电路设计^[7,8]、生命医学^[9]等诸多领域有着广泛的应用. 同时混沌系统行为轨迹非常复杂, 其运动轨迹难以预测, 近年来混沌系统控制成为非线性系统控制的热点, 引起了广大学者的关注, 相继提出了诸多有效的控制方法, 如预测控制^[10]、智能PID控制^[11]、自适应控制^[12,13]、模糊控制^[14,15]、以及基于神经网络控制^[16]等方法. 现有控制方法大多需要精确系统模型, 又现实非线性系统往往存在有扰动且模型不确定性因素, 系统模型难以确定.

自抗扰控制器(active disturbance rejection control, ADRC)是一种不依赖于模型的非线性控制器, 具有精度高、响应速度快、抗干扰能力强等特点^[17], 近年来业已广泛应用于非线性系统控制中^[18], 但对实时动态系统控制有所欠缺. 小脑模型神经网络^[19](cerebellar model articulation controller, CMAC), 是仿照小脑控制肢体运动原理而建立的神经网络模型, 具有很强的局部逼近能力、和一定的

泛化能力, 学习速度快, 广泛应用于实时系统控制及非线性系统控制等领域^[20], 且能够克服ADRC在控制非线性系统时动态性能差的缺点. 但神经网络控制的稳定性受到迭代初值和训练程度的影响, 控制初期效果不佳, 抗干扰能力不强.

针对受扰不确定离散非线性混沌系统, 利用ADRC与CMAC各自的优势, 将自抗扰控制技术与CMAC神经网络控制相结合, 提出一种自抗扰神经网络并行优化控制方法(ADRC-CMAC)试图将离散混沌系统变成期望规则运动. 利用免疫动态微粒群算法(immune dynamic particle swarm optimization, IDPSO)对ADRC-CMAC控制器参数进行自学习寻优, 使控制器参数得到最优配置. 文中针对一类受扰离散非线性系统及离散混沌系统研究结果表明, 该控制方法具有更好的动态性能和较强的鲁棒性.

2. 问题描述

考虑如下离散非线性混沌动态系统:

$$y(k+1) = f(y(k), y(k-1), \dots, y(k-m), u(k)),$$

* 国家自然科学基金重点项目(批准号:60634020), 湖南省自然科学基金(批准号:07JJ3126), 湖南省科技计划重点项目(批准号:2007GK3078)资助的课题.

† E-mail: zhangyj@hnu.edu.cn, zhaohualiu2009@hotmail.com

$$u(k-1), \dots, u(k-n)) + f_N(k), \quad (1)$$

其中 $y(k), u(k) \in R$ 分别为 k 时刻的系统输出、输入信号, $f_N(\cdot)$ 表示未知动态、扰动或其他参数不确定性等因素. $m, n \in R$ 为系统输出, 输入阶数, $f(\cdot)$ 是未知非线性函数.

3. ADRC 与 CMAC 并行控制算法

3.1. 自抗扰控制

自抗扰控制器^[17]是针对一类非线性不确定系统提出的一种非线性控制器(图1),汲取了经典PID算法精华.其由跟踪微分器(TD),扩张状态观测器(ESO)和非线性状态误差反馈律(NLSEF)三部分构成.

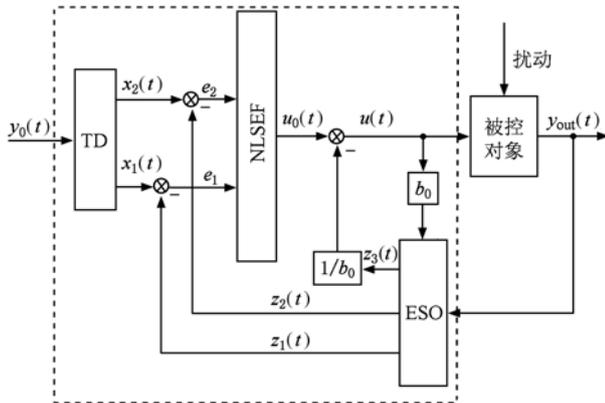


图1 自抗扰控制器(ADRC)的结构

3.1.1. 跟踪微分器

跟踪微分器一个动态环节,对其输入信号,它将给出两个输出,其中一个跟踪输入信号,另一个是输入信号的导数,有效抑制信号噪声干扰.

$$\begin{aligned} x_1(k+1) &= x_1(k) + Tx_2(k), \\ x_2(k+1) &= x_2(k) + Tfst(x_1(k), \\ &\quad x_2(k), r, h_0), \end{aligned} \quad (2)$$

(2)式中, T 为采样步长, r 为决定跟踪快慢的参数, h_0 为滤波因子,非线性函数 fst 为

$$fst = - \begin{cases} ra/d, & |a| \leq d, \\ r\text{sgn}(a), & |a| > d, \end{cases} \quad (3)$$

式中的参数变量关系如下:

$$\begin{aligned} d &= rh_0, \\ d_0 &= dh_0, \\ y(k) &= x_1(k) - y_0 + h_0x_2(k), \end{aligned}$$

$$\begin{aligned} a_0 &= \sqrt{d^2 + 8r|y(k)|}, \\ a &= \begin{cases} x_2 + \text{sign}(y(k))(a_0 - d)/2, & |y| > d_0, \\ x_2 + y(k)/h_0, & |y| \leq d_0. \end{cases} \end{aligned} \quad (4)$$

3.1.2. 扩张状态观测器

采用非线性结构将系统的状态和模型的不确定及外部扰动进行估计.

$$\begin{aligned} e(k+1) &= z_1(k) - y(k+1), \\ z_1(k+1) &= z_1(k) + T[z_2(k) - \beta_{01}e(k)], \\ z_2(k+1) &= z_2(k) + T[z_3(k) \\ &\quad - \beta_{02}\text{fal}(e, 0.5, \delta) + b_0u(k)], \\ z_3(k+1) &= z_3(k) - T\beta_{03}\text{fal}(e, 0.25, \delta), \end{aligned} \quad (5)$$

式中, $\beta_{01}, \beta_{02}, \beta_{03}$ 为输出误差校正增益可调参数,非线性函数 $\text{fal}(\cdot)$ 为

$$\text{fal}(e, \alpha, \delta) = \begin{cases} |e|^{\alpha} \text{sgn}(e), & |e| > \delta, \\ e/\delta^{1-\alpha}, & |e| \leq \delta, \end{cases} \quad \delta > 0. \quad (6)$$

3.1.3. 非线性状态误差反馈律

把传统的PID“线性”和形式转为“非线性组合”,得到非线性PID,提高系统的控制品质.

$$\begin{aligned} e_1(k+1) &= x_1(k+1) - z_1(k+1), \\ e_2(k+1) &= x_2(k+1) - z_2(k+1), \\ u_0(k+1) &= \beta_1 \text{fal}(e_1(k+1), \alpha_1, \delta) \\ &\quad + \beta_2 \text{fal}(e_2(k+1), \alpha_2, \delta). \end{aligned} \quad (7)$$

自抗扰控制器总的输出为

$$u_{\text{adrc}}(k+1) = u_0(k+1) - z_3(k+1)/b_0, \quad (8)$$

其中 β_1, β_2, b_0 为可调参数.

3.2. CMAC 结构

CMAC网络^[19]由输入层、中间层和输出层组成. CMAC的输入输出非线性关系由概念映射($U \rightarrow AC$)和实际映射($AC \rightarrow AP$)来实现,其结构如图2所示.

3.3. ADRC 与 CMAC 并行控制算法

该控制系统由ADRC和CMAC神经网络组成. ADRC实现反馈控制根据系统的输出误差,产生反馈信号,保证控制过程中系统的稳定性,且抑制系统的扰动;CMAC通过学习实现被控对象的逆动态模型,在控制初期对ADRC进行学习,对系统进行实时动态跟踪;动态免疫微粒群算法(IDBPSO)对ADRC-CMAC控制器参数进行全局优化,以保证ADRC-CMAC控制器参数得到最优配置.

系统的控制算法为

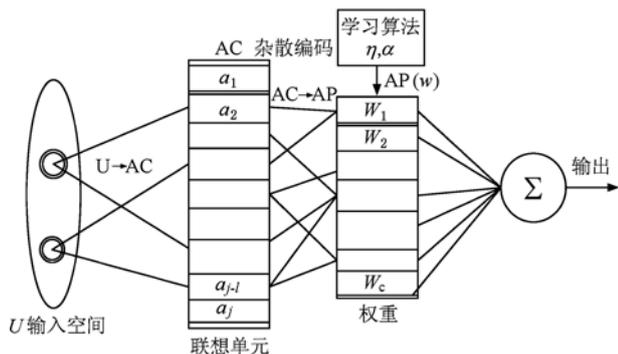


图2 CMAC 结构图

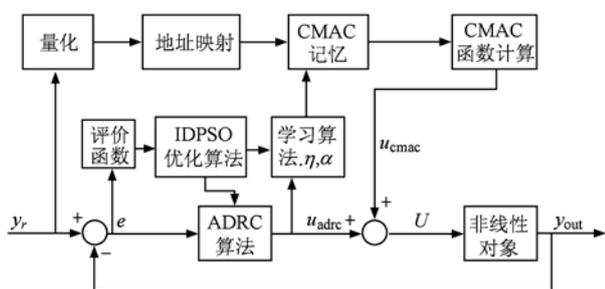


图3 ADRC 与 CMAC 并行优化控制结构图

$$u_{\text{cmac}}(k) = \sum_{i=1}^c w_i a_i, \quad (9)$$

$$U(k) = u_{\text{adrc}}(k) + u_{\text{cmac}}(k), \quad (10)$$

(9)式中, a_i 为二进制选择向量, c 为 CMAC 网络的泛化参数; (10)式中 $u_{\text{cmac}}(k)$ 为 CMAC 产生相应的输出, $u_{\text{adrc}}(k)$ 为 ADRC 控制器产生的输出, $U(k)$ 为总控制器输出。

CMAC 采用有导师的学习算法. 每一控制结束时, 计算出相应的 CMAC 输出 $u_{\text{cmac}}(k)$, 并与总控制器输入 $u(k)$ 相比较, 修正权重, 进入学习过程。

CMAC 的调整目标为

$$E(k) = \frac{1}{2}(u_n(k) - u(k))^2 \cdot \frac{1}{c}, \quad (11)$$

$$\begin{aligned} \Delta w(k) &= -\eta \frac{\partial E(k)}{\partial w} = \eta \frac{u(k) - u_n(k)}{c} a_i \\ &= \eta \frac{u_a(k)}{c} a_i, \end{aligned} \quad (12)$$

$$\begin{aligned} w(k) &= w(k-1) + \Delta w(k) \\ &+ \alpha(w(k) - w(k-1)). \end{aligned} \quad (13)$$

当系统开始运行时, 系统由 ADRC 控制器进行控制, 保证系统的快速性及稳定性. 通过 CMAC 的不断学习及随着系统误差信号逐渐减小, 使 ADRC

产生的输出控制量 $u_{\text{adrc}}(k)$ 逐渐减弱, 直到 CMAC 产生的输出控制量 $u_{\text{cmac}}(k)$ 逐渐逼近控制器。

4. 免疫动态微粒群优化方法

由以上分析可以可知 ADRC 控制器参数 $\{\beta_{01}, \beta_{02}, \beta_{03}, \beta_1, \beta_2, b_0\}$ 及 CMAC 参数 $\{\eta, \alpha\}$ 构成的参数集合 $\{\beta_{01}, \beta_{02}, \beta_{03}, \beta_1, \beta_2, b_0, \eta, \alpha\}$ 需要整定. 本文利用免疫动态微粒群算法对 ADRC-CMAC 参数进行优化. 免疫动态微粒群算法其原理如下。

4.1. 动态微粒群机理

粒子群优化算法 PSO (particle swarm optimization) 源于对鸟类等微粒捕食行为的模拟, 最初由 Eberhart 和 Kennedy^[20] 提出的一种仿生进化算法, 并广泛的应用于科学与工程领域^[21]. 然而, 微粒群算法与其他进化算法类似, 也是基于群体进化, 在进化后期容易陷入早熟收敛. 为了保证群体的多样性及强搜索能力; 设计群体自适应分群算子, 在群体进化过程中微粒群分成两种不同行为特征子群, 探索群体与捕食群体. 两分群体按照不同的搜索方式进行优化, 其群体行为可以用简单的自动机模型表示见图 4。

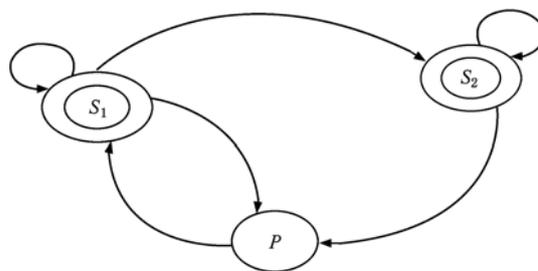


图4 动态微粒群自动机模型(S_1 : 捕食状态, S_2 : 探索状态, P 为群体)

由图 4 可知动态行为表明群体最初处于捕食状态, 在捕食状态下搜索速度减慢或停止, 群体中一部分微粒将从捕食状态转为探索状态以扩大解空间. 同时群体也可以从探索状态归队转为捕食状态进行集中优良区域搜索加快搜索速度, 每一个状态都可能收敛到全局最优值. 两分群微粒归队合并以后对微粒进行更新, 并对群体微粒按照亲和度高低排列。

1) 捕食状态: 在捕食状态下微粒群行为与传统的微粒群算法一致. 设粒子群体规模为 N , 每个粒子在 D 维搜索空间中, 记粒子 i ($i = 1, 2, \dots, N$) 的当前

位置是 $X_i = \{X_{i1}, X_{i2}, \dots, X_{id}\}$, 飞行的速度为 $V_i = \{V_{i1}, V_{i2}, \dots, V_{id}\}$ 每个粒子当前找到的最优解为 P_{id} , 种群当前找到的全局最优解为 P_{gd} . 通常限定 $V_{\min} \leq V_{id} \leq V_{\max}$.

$$V_{id}(t+1) = \omega V_{id} + c_1 \text{rand}() (P_{id}(t) - X_{id}(t)) + c_2 \text{rand}() (P_{gd}(t) - X_{id}(t)), \quad (14)$$

ω 为惯性权重. 一般随着进化线性减少, c_1, c_2 为调节 P_{id} 和 P_{gd} 相对重要性的参数, $\text{rand}()$ 为生成介于 0 和 1 之间的随机数. 这样, 可以得到粒子的下一位置:

$$X_{id}(t+1) = X_{id}(t) + V_{id}(t+1). \quad (15)$$

2) 探索状态: 在优化过程中, 微粒群根据群体当前所在的状态, 如果全局最优解且优化能力减慢, 则整个群体已陷入局部极值点, 依据自适应动态分群算子子群体自适应进行动态分群将一部分微粒群转为探索状态, 探索群体有能力探索有希望的区域以寻求更优的解, 同时又不失群体的多样性. 自适应分群算子定义如下:

$$f = \frac{\text{avgfit}_i - \text{avgfit}_{i-1}}{\text{avgfit}_{i-1} - \text{avgfit}_{i-2} + \delta}, \quad (16)$$

avgfit_i 为第 i 代种群平均适应度值, $\delta \in (0, 1)$ 为平滑系数, 防止分母为 0. (10) 式表明, 当 $f=0$ 时表明群体进化停止; 当 $0 < f < 1$ 时, 表示种群进化速度减慢; 当 $f > 1$ 时进化加速. 当 $f < 1$ 时以 $(1-f)$ 的概率进行探索. 当进行探索时, 保留性能最好一部分微粒在捕食状态, 其余微粒转为探索状态.

当微粒从捕食状态转化为探索状态时, 微粒将重新随机初始化分布在搜索空间.

$$X_{kd}(t) = X_{kd}^{\min} + \text{rand}() (X_{kd}^{\max} - X_{kd}^{\min}), \quad (17)$$

其速度位置更新公式如下:

$$V_{kd}(t+1) = \omega \cdot \text{sign}(r) \cdot V_{kd} + c_1 \cdot \text{rand}() \times (P_{kd}(t) - X_{kd}(t)) + c_2 \cdot \text{rand}() \times (P_{gd}(t) - X_{kd}(t)) + c_3 \cdot u(0, 1) \times (X_{kd}(t) - hP_{gd}), \quad (18)$$

$$X_{kd}(t+1) = X_{kd}(t) + V_{kd}(t+1), \quad (19)$$

$$\text{sign}(r) = \begin{cases} 1, & (r \geq 0.05), \\ -1, & (r < 0.05), \end{cases} \quad (20)$$

c_3 为 $(0, 1)$ 之间的随机数, r 为随机数, $\text{sign}(r)$ 为符号函数, 为了寻求更优目标, 微粒可能改变飞行方向, $u(0, 1)$ 为高斯分布函数. hP_{gd} 为当前探索种群中最优粒子在第 d 维的位置.

3) 精英粒子局部增强算子: 精英粒子为群体领导者. 为防止精英微粒陷入局部极值点, 本文设计

一种局部增强算子以协助部分精英微粒跳出局部极值点

$$G_{kd} = \text{gbest} + m(X_{r1d} - X_{r2d}) \times \left(1 - \frac{t}{T}\right), \quad (21)$$

式中 G_{kd} 为增强后的微粒, gbest 为当前代全局最优个体, m 为限幅常数, t 为代数, T 为算法执行总代数, 且 $r_1 \neq r_2 \neq k, X_{r1d}, X_{r2d}$ 随机选取. 增强算子促使这部分微粒在当前种群中的最优个体附近进行随机微调便于逃离局部点. 增强算子以增强算法局部搜索能力, 而精英粒子的最优信息不断向种群其它微粒传播加快了算法的收敛速度.

4. 2. 人工免疫算法

进化后期多数的粒子易集中收敛于局部极值点, 失去多样性, 且粒子自身没带有适应性变异能力, 致使群体后期优化能力下降很难找到全局最优极值点. 引入人工免疫系统^[22] 中克隆选择机理和疫苗接种机理, 增强微粒群多样性、大范围变异能力及收敛速度.

4. 2. 1. 克隆选择

克隆操作步骤如下.

步骤 1 各个粒子的个体极值 $\{P_{1d}, P_{2d}, \dots, P_{Nd}\}$ 组成一个临时的克隆种群. 将临时克隆种群每一粒子视为抗体, 选择 $k (k < N)$ 个最好的抗体个体组成子群体 sub_1 , 对子群体 sub_1 进行克隆复制, 公式如下:

$$\text{sub}_1(i) = \text{round}\left(\frac{\beta N}{i} + b\right), \quad (22)$$

其中 N 为种群规模. $\beta \in (0, 1)$. 保证每个抗体都有一定克隆数量, 故加上了常量 $b \geq 1$ 的整数. 经过克隆扩增生成新群体 sub_2 .

步骤 2 对群体 sub_2 中的每个个体实施高频变异, 其方法为基于时间变尺度的邻域内变异算子其变异公式如下:

$$P_{id}^{\text{new}} = P_{id} + [(r_d > P_m)] \eta P_{id} U(0, 1) - [(r_d \leq P_m)] \eta P_{id} U(0, 1), \quad (23)$$

$$[r_d > P_m] = \begin{cases} 1, & r_d > P_m, \\ 0, & \text{其它}, \end{cases} \quad (24)$$

r_d 为随机数, $P_m = 0.5$,

$$\eta(t) = 1 - r^{[1 - (t/T)]^b}. \quad (25)$$

由上式可知算法在进化初期以一定的变异概率采用较大的变异尺度以保持种群的多样性, 而在进化

后期变异尺度逐渐缩小以提高局部微调能力。

步骤3 免疫选择操作,从克隆变异后的个体中选择亲和力最优的个体进入下一代。

4.2.2. 疫苗接种

疫苗接种是利用优秀疫苗抗体信息来改变抗体的部分结构以提高抗体对抗原的识别能力,从而提高抗体群的性能.在进化过程中每隔数代从当前群体亲和力最优的个体中提取疫苗,对亲和力不高的粒子进行自适应地疫苗接种.其接种表达式为

$$p'_{kd} = p_{kd} + \sigma(p_{bestkd} - p_{kd}), \quad (26)$$

$$\sigma = \sigma_{max} - (\sigma_{max} - \sigma_{min}) \frac{t}{T}, \quad (27)$$

式中 p_{kd} 为需接种的微粒, p_{bestkd} 为较优的微粒, $\mu = 0$, t 是群体当前进化代数, T 为总的进化代数, σ 为学习率线性递减, $\sigma_{max} = 1.0$, $\sigma_{min} = 0.1$. 若接种后的个体亲和力不如接种前的粒子,则取消疫苗接种。

4.3. 基于 IDPSO 的 ADRC - CMAC 优化设计流程

设计流程如图 5 所示。

5. 实验

5.1. 数值仿真实例

为了验证文中控制算法的强抗干扰能力,选取一类时刻强正弦扰动离散非线性系统进行测试,并与最近算法进行比较,其系统模型来自文献[23],

$$y(k+1) = \frac{y(k)}{1 + y(k)^2 + d(k)} + u(k)^3. \quad (28)$$

期望信号: $y_r(k+1) = 1$;

扰动信号: $d(k) = \sin(k/50)$.

表1 五种算法性能指标比较

P	NN-SMC1	NN-SMC2	NN-SMC3	BP-PID	ADRC-CMAC
$\delta/\%$	2.27	2.13	2.06	4.26	1.79
J	6.67	4.65	4.57	11.54	4.00

表1中 NN-SMC 为提出一种基于神经网络估计的非线性离散时间系统自适应准滑模控制方法^[23].

(NN-SMC1 为基于紧格式、NN-SMC2 偏格式和 NN-SMC3 全格式线性化得出的三种 NN-SMC 控制律).

BP-PID 为基于 BP 网络整定的 PID 控制算法^[24]. δ

表示系统超调量, $J = \int_0^t |y_r - y_{out}| dt$ 为绝对误差

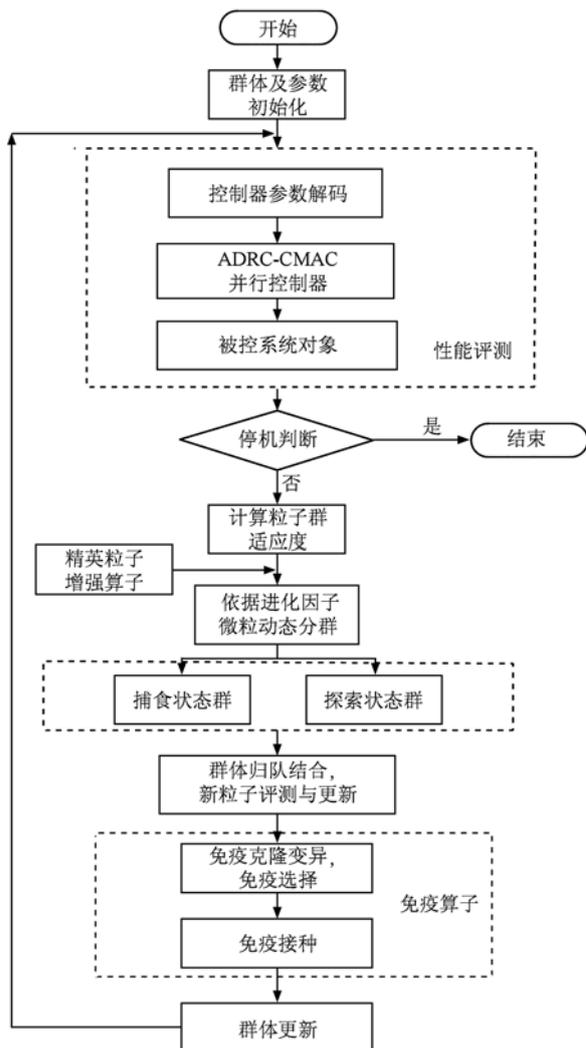


图5 基于免疫动态微粒群的 ADRC-CMAC 并行优化控制

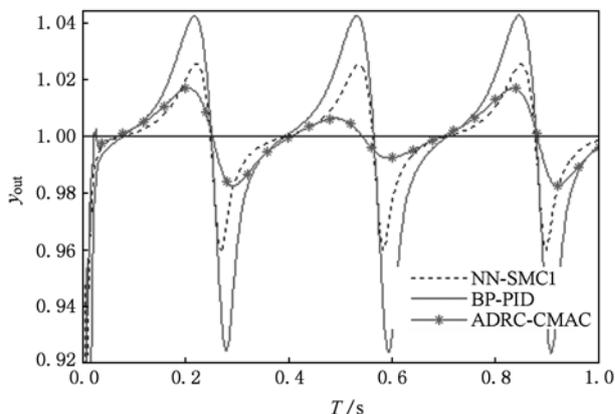


图6 三种方法的比较

的一阶积分。

通过图 6 和表 1,可以清楚地看到本文 ADRC-

CMAC 方法在抗扰动、超调量及误差绝对值等各项性能指标上均优于 NN-SMC1, NN-SMC2, NN-SMC3 及 BP-PID 方法.

5.2. 不确定 Henon 混沌系统控制

Henon 混沌系统方程^[10,11]为

$$\begin{aligned} x_1(k+1) &= 1 - (a + \sigma a)(x_1(k))^2 \\ &\quad + (b + \sigma b)x_2(k) + \sigma d_1 + u(k) \\ x_2(k+1) &= x_2(k) + \sigma d_2. \end{aligned} \quad (29)$$

当 $u(k) = 0$ 时系统处于混沌态. 其中不确定性为 $\sigma a = 0.02$, $\sigma b = 0.03$, 外部干扰为 $\sigma d_1 =$

$0.1\sin(k)$, $\sigma d_2 = 0.01\sin(k)$, 其不稳定平衡点为 $(0.6314, 0.6314)$, 为了验证本文控制算法的有效性在 $n=30$ (第 30 步) 时对系统加入控制量. 控制到平衡态. 从图 7(a) 可以看出混沌系统在内部参数摄动与外部扰动的情况下, 本文 ADRC-CMAC 控制方法很快控制到平衡点, 具有很强的抗扰能力; 由图 7(b) 可知控制初期主要由 ADRC 控制器起作用, 后期作用减弱且有微微的震动, 但一直起到抑制系统干扰作用, CMAC 不断学习迅速逼近系统, 动态性能较好; 由图 7(d) 控制适应度曲线收敛速度较快, 说明文中提出的免疫动态微粒群算法全局寻优能力强.

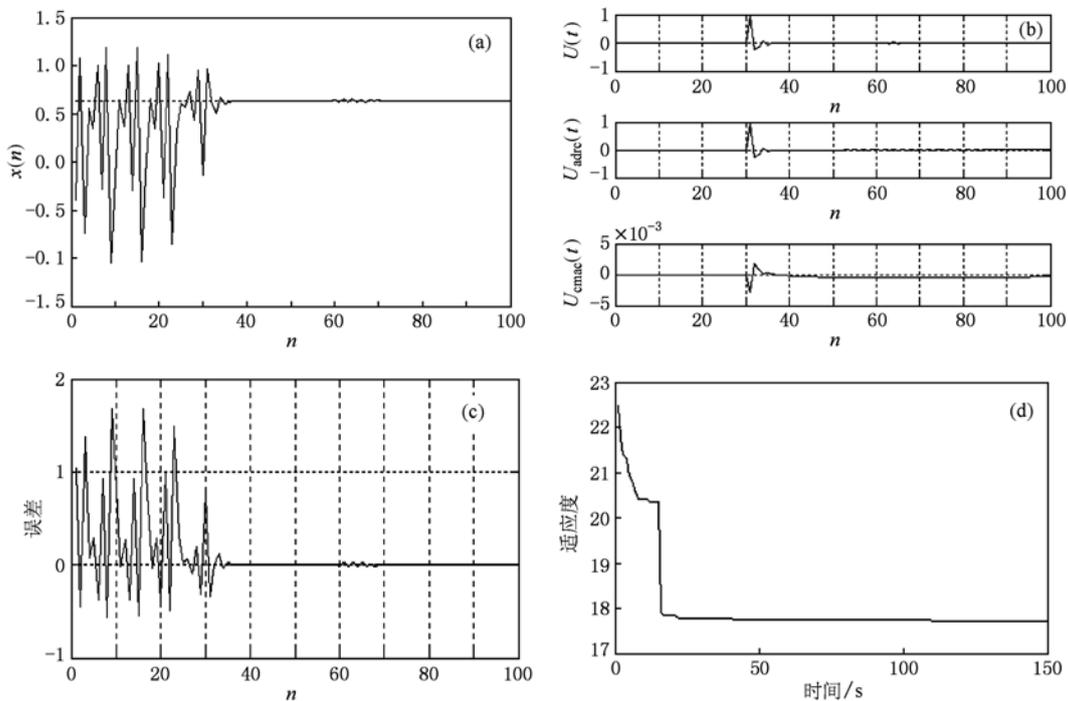


图 7 不确定 Henon 混沌系统控制 (a) 系统输出; (b) 各个控制器的输出; (c) 系统误差; (d) 适应度变化曲线

5.3. 六辊 UC 轧机混沌系统控制

5.3.1. 无噪声环境下的轧机混沌系统控制

六辊 UC 轧机混沌系统^[11]模型为

$$x(n+1) = rx(n)\exp(1 - x(n)^2) + u(n). \quad (30)$$

此系统未施加控制作用, 当 $U(n) = 0$ 时, 在 $r \geq 1.5$ 的情况下, 系统处于混沌态. 初始条件为 $x(0) = 1.45$, 系统平衡点为 1.199273. 而工程实际中所要求的控制目标往往不在系统的平衡点, 能够控制系统到任一非平衡点这是混沌系统控制的一大难题, 为了验证算法的有效性, 当 $n \geq 50$ 时加入

控制作用, 控制到非平衡态, 取给定恒值为 $x_r(n) = 2$, 从图 8 可以看出 ADRC-CMAC 控制方法可以控制混沌系统到某一给定恒值.

5.3.2. 噪声环境下的轧机混沌系统控制

带扰动的六辊 UC 轧机混沌系统模型如下:

$$x(n+1) = rx(n)\exp(1 - x(n)^2) + u(n) + \xi(k), \quad (31)$$

其中 $\xi(k)$ 为均值为 0, 方差为 0.01 的高斯随机噪声. 当 $n \geq 50$ 时加入控制作用, 控制到非平衡态, 取给定恒值为 $x_r(n) = 2$, 从图 9 可以看出 ADRC-CMAC 控制方法在混沌系统外加扰动的情况下仍然能对系统进行有效的控制.

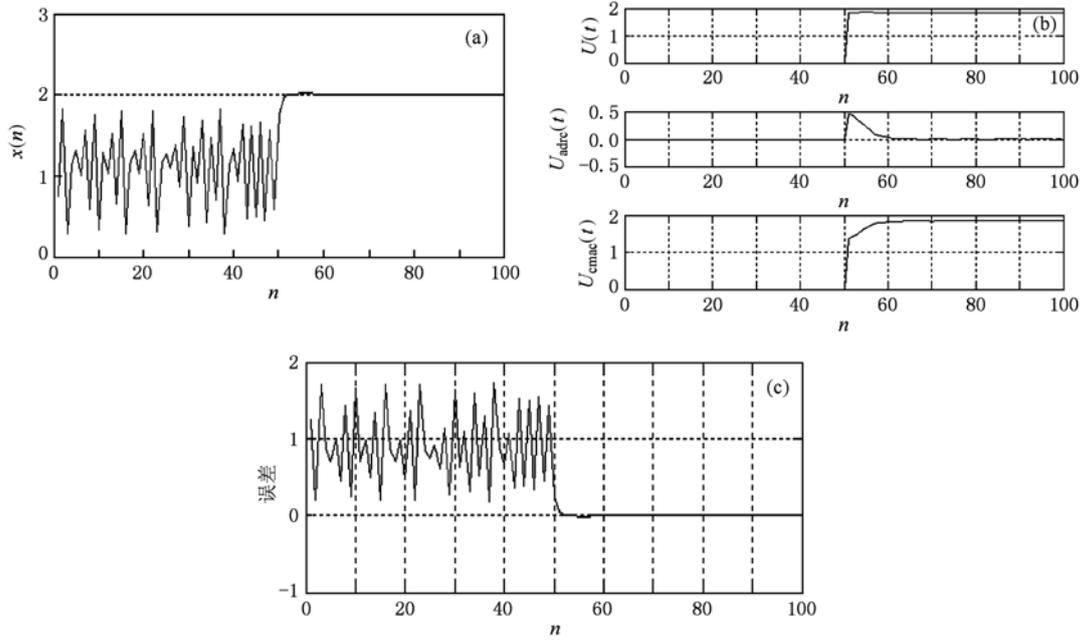


图8 给定值下六辊 UC 轧机混沌系统控制 (a) 系统输出; (b) 各个控制器的输出; (c) 系统误差

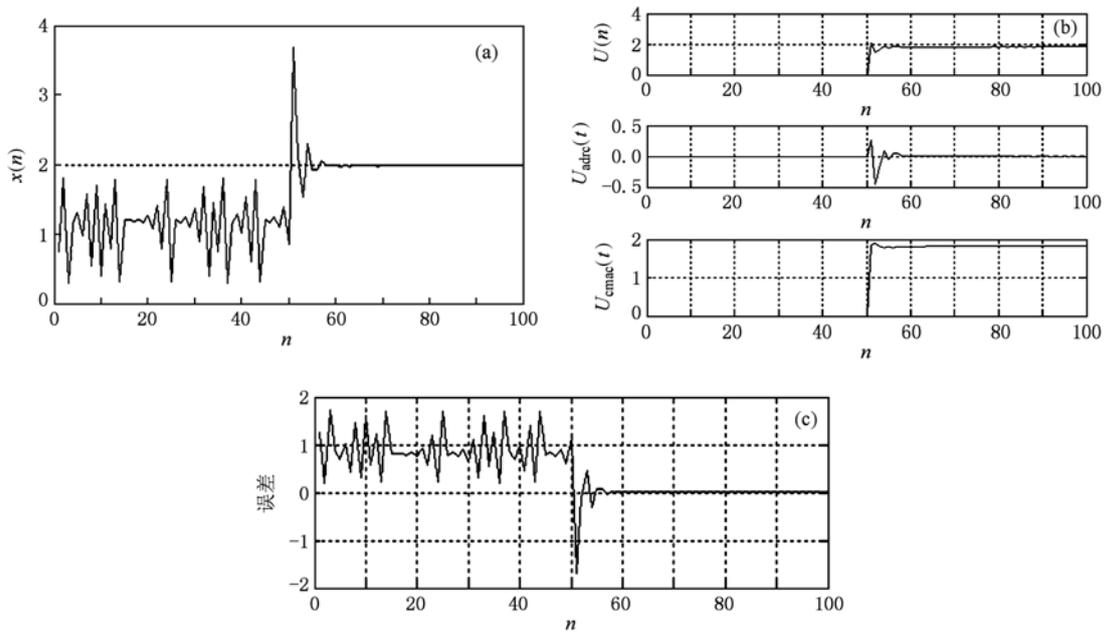


图9 噪声环境下六辊 UC 轧机混沌系统控制 (a) 系统输出; (b) 各个控制器的输出; (c) 系统误差

从以上实验结果可以看出控制初期主要由 ADRC 控制器起作用,抑制干扰,CMAC 控制器对 ADRC 快速学习,加快控制响应速度,后期 CMAC 控制器起主要作用. 同时在干扰的环境下,ADRC 能有效的抗干扰,CMAC 控制器快速的实现跟踪. IDBPSO 算法对 ADRC-CMAC 参数全局寻优,使得效果得到大幅度提高.

6. 结 论

针对一类受扰不确定离散非线性混沌系统,提出了一种 ADRC 与 CMAC 神经网络并行控制方法 (ADRC-CMAC),利用免疫动态微粒群算法对控制

器参数进行寻优,有效避免了寻优过程陷入局部极小问题,从而实现了系统的优化控制. 对一类受扰离散非线性系统、不确定 Henon 混沌系统及六辊 UC 轧机混沌系统进行控制仿真结果证明,该算法

在控制品质上有很大的提高,在系统动态响应速度、稳定性、抗干扰能力等方面都表现出很好的性能. 为离散非线性混沌系统控制提供一种新的智能控制方法.

- [1] Li W, Hao J H, Q B 2008 *Acta Phys. Sin.* **57** 1398 (in Chinese) [李伟、郝建红、祁兵 2008 物理学报 **57** 1398]
- [2] Xu S J, Wang J Z 2008 *Acta Phys. Sin.* **57** 37 (in Chinese) [徐淑奖、王继志 2008 物理学报 **57** 37]
- [3] Yang J Y, Liao X F, Xiao D, Deng S J 2008 *Journal on Communications* **29** 86 (in Chinese) [杨吉云、廖晓峰、肖迪、邓绍江 2008 通信学报 **29** 86]
- [4] Wang L P, Liu W, Xiang H S 2008 *IEEE Transactions on Systems, Man, and Cybernetics, C: Applications and Reviews* **38** 209
- [5] Jovic B, Unsworth C P 2010 *Electronics Letters* **46** 49
- [6] Pisarchik A N, Ruiz-Oliveras F R 2010 *IEEE Journal of Quantum Electronics* **46** 279
- [7] Zhou P, Cao Y X, Cheng X F 2009 *Chin. Phys. B* **18** 1394
- [8] Li Y, Zhang Z M, Tao Z J. 2009 *Acta Phys. Sinica* **58** 6818 (in Chinese) [李亚、张正明、陶志杰 2009 物理学报 **58** 6818]
- [9] Ning X B, Bian C H, Wang J 2006 *Chinese Science Bulletin* **51** 385 (in Chinese) [宁新宝、卞春华、王俊 2006 科学通报 **51** 765]
- [10] Wen S H 2009 *Acta Phys. Sin.* **58** 5209 (in Chinese) [温淑焕 2009 物理学报 **58** 5209]
- [11] Wang D F, Han P 2006 *Acta Phys. Sin.* **55** 1644 (in Chinese) [王东风、韩璞 2006 物理学报 **55** 1644]
- [12] Antonio L, Arturo Z R 2007 *IEEE Transactions on Circuits and Systems I: Regular Papers* **54** 2019
- [13] Alexander L F, Boris A, Robin J E 2008 *IEEE Trans. Circ. Syst.* **55** 1685
- [14] Lam H K, Ling W K, Herbert H C I, Steve S. H. Ling, Member 2008 *IEEE Transactions on Circuits and Systems I: Regular Papers* **55** 893
- [15] Wang X Y, Meng J 2009 *Acta Phys. Sin.* **58** 3780 (in Chinese) [王兴元、孟娟 2009 物理学报 **58** 3780]
- [16] Juang C F, Huang R B, Lin Y Y 2009 *IEEE Transactions on Fuzzy Systems* **17** 1092
- [17] Han J Q 2009 *IEEE Transactions on Industrial Electronics* **56** 900
- [18] Han J Q 2008 *Active Disturbance Rejection Control Techniqu* (Beijing: Natinal Defense Industry Press p5) (in Chinese) [韩京清 2008 自抗扰控制技术 (北京:国防工业出版社) 第5页]
- [19] Albus J S 1975 *Transactions of the ASME Journal of Dynamic System Measurement and Control* **97** 220
- [20] Lin C M, Chen T Y 2009 *IEEE Transactions on Neural Networks* **20** 1377
- [21] Eberchart R. Kennedy J 1995 *Proc. Sixth International Symposium on Micro Machine and Human Science, Nagoya, Japan* 39
- [22] Dasgupta D 2006 *IEEE Computational Intelligence Magazine* **1** 40
- [23] Hou Z S, Wang W H, Jin S T 2009 *Control Theory & Applications* **26** 505 (in Chinese) [侯忠生、王卫红、金尚泰 2009 控制理论与应用 **26** 505]
- [24] LIU J K 2007 *Advanced PID control and MATLAB Simulation* (Beijing: Eleectrical Industry Press) p163, p263 (in Chinese) [刘金琨 2007 先进PID控制 MATLAB 仿真 (电子工业出版社) 第163, 263页]

ADRC and CMAC combined optimization and control for a class of discrete-time uncertain chaotic systems^{*}

Liu Zhao-Hua^{1)2)†} Zhang Jing¹⁾ Zhang Ying-Jie²⁾ Li Xiao-Hua³⁾ Wu Jian-Hui²⁾

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

2) (School of Computer and Communication, Hunan University, Changsha 410082, China)

3) (Department of Electronic and Information Engineering, Changsha Normal College, Changsha 410100, China)

(Received 30 May 2010; revised manuscript received 3 July 2010)

Abstract

Immune dynamic particle swarm optimization (IDPSO) strategy integrated with active disturbance rejection control (ADRC) and cerebellar model articulation controller (CMAC) combined control is designed for uncertain nonlinear discrete-time chaotic systems. The ADRC-CMAC is comprised of a cerebellar model articulation controller (CMAC) and an ADRC controller. The ADRC controller is designed to guarantee the stability of the system and restrict the disturbance. The CMAC is used to guarantee the control precision and response speed. Immune binary-state particle swarm algorithm is used to tune online the parameters of the ADRC-CMAC. Simulation results of uncertain nonlinear discrete-time systems demonstrate that performance with favorable response speed and restrained disturbance can be achieved by using the proposed control system.

Keywords: active disturbance rejection control (ADRC), cerebellar model articulation controller (CMAC), parallel control, chaotic system

PACS: 07.05.Mh, 95.10.Fh, 02.30.Yy, 05.45.Gy

^{*} Project supported by the Key National Natural Science Foundation of China (Grant No. 60634020), the Hunan Provincial Natural Science Foundation of China (Grant No. 07JJ3126), the Planned Science and Technology Project of Hunan Province, China (Grant No. 2007GK3078).

[†] E-mail: zhangyj@hnu.edu.cn, zhaohualiu2009@hotmail.com