

# 基于混沌粒子群优化算法的异结构混沌反同步自抗扰控制

刘福才<sup>†</sup> 贾亚飞 任丽娜

(燕山大学西校区, 工业计算机控制工程河北省重点实验室, 秦皇岛 066004)

(2012年7月17日收到; 2013年2月15日收到修改稿)

针对一类连续时间异结构混沌系统, 利用自抗扰控制很强的鲁棒性, 提出了一种异结构混沌系统反同步的自抗扰控制策略。针对所设计的自抗扰控制器参数较多, 难以整定的问题, 提出了应用混沌粒子群优化算法对控制器进行参数寻优设计。以 Lorenz 系统和 Chua 系统两个异结构混沌系统为例进行仿真验证, 由仿真结果可知, 该方法可以实现异结构混沌系统较快的反同步控制, 且具有很强的抗干扰能力。

**关键词:** 异结构混沌系统反同步, 自抗扰控制器, 混沌粒子群优化算法, 参数寻优

**PACS:** 05.45.Gg, 05.45.Ac, 05.45.Pq

**DOI:** 10.7498/aps.62.120509

## 1 引言

近年来, 混沌同步在图像加密、保密通信和信息工程等领域应用广泛, 推动了混沌同步理论的快速发展。同时, 在周期混沌系统中观察到了反同步现象。混沌反同步是指达到同步的两个混沌系统的状态向量的绝对值相同但符号相反<sup>[1]</sup>。混沌反同步控制研究已经开始引起学者的关注。而在实际应用中, 由于完全相同的两个混沌系统几乎是不存在的, 因此不同混沌系统即异结构混沌系统的反同步问题的研究更具实际意义。目前, 已经出现了多种异结构混沌反同步的控制方法, 例如基于主动滑模控制的异结构混沌系统反同步控制<sup>[1]</sup>, 基于自适应方法的双向耦合混沌系统反同步控制<sup>[2]</sup>, 基于反对称结构直接构造法以及主动控制法的不同结构超混沌系统反同步控制<sup>[3]</sup>, 基于指数函数控制方法的异结构混沌系统反同步控制<sup>[4]</sup>, 基于状态观测器和极点配置方法的混沌系统反同步控制<sup>[5]</sup>, 基于自适应主动及滑模控制的分数阶超混沌系统异结构反同步控制<sup>[6]</sup>, 基于 Lyapunov 稳定性理论和自适应控制方法的参数不确定混沌系统反同步控制<sup>[7]</sup>等。但是, 上述方法多存在算法复杂、需要精确的被控

对象的数学模型、不能很好抑制系统可能出现的各种内外扰动等缺点, 在实际应用中存在很大的局限性。因此, 本文提出一种异结构混沌系统反同步自抗扰控制器。

自抗扰控制器 (ADRC)<sup>[8]</sup> 是一种不依赖于对象精确数学模型的新型非线性鲁棒控制技术。它利用状态扩张观测器 (ESO) 观测系统内外扰动以及系统建模误差, 并通过误差反馈 (NLSEF) 对系统扰动进行补偿, 具有很强的鲁棒性和模型适应性, 算法简单, 并且已经成功应用到了一些工业领域。但是, ADRC 需要整定的参数很多, 目前其参数的整定过程和效果很大程度上还依赖于人们的经验, 整定过程繁琐, 具有很多不确定性。目前已经出现了多种自抗扰控制器参数的整定方法, 主要是结合各种智能算法对 ADRC 参数进行寻优, 如基于遗传算法的 ADRC 参数整定算法<sup>[9]</sup>, 基于免疫双态微粒群的 ADRC 参数整定算法<sup>[10]</sup>, 基于小生境粒子群的 ADRC 参数整定算法<sup>[11]</sup>等。本文提出一种基于混沌优化思想的混沌粒子群优化 (CPSO) 算法的自抗扰控制器。CPSO<sup>[12,13]</sup> 算法是基于标准粒子群算法的基础上采用混沌系列初始化粒子的位置和速度提高了种群的多样性和粒子搜索的遍历性, 优化过

<sup>†</sup> 通讯作者。E-mail: lfc@ysu.edu.cn

程中利用混沌变量进行优化搜索避免陷入局部最优值,因此具有较强的全局搜索能力、较快的收敛速度以及较高的精度.将 CPSO 算法用于自抗扰控制器参数优化,应用于异结构的混沌反同步控制中,构建了一种基于混沌粒子群算法的自抗扰控制器(CPSO-ADRC),使两个异结构的混沌系统实现反同步的控制要求.仿真结果表明,本文所提方法可以有效提高自抗扰控制器对异结构混沌系统反同步的控制能力,并且对系统扰动具有很好的补偿作用,取得了较好的控制效果.

## 2 异结构混沌反同步问题描述

设两个混沌系统如下:

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{F}(\mathbf{X}, t), \\ \dot{\mathbf{y}} = \mathbf{H}(\mathbf{Y}, t) + \mathbf{G}, \end{cases} \quad (1)$$

其中第一个为驱动系统,第二个为响应系统.  $\mathbf{X}, \mathbf{Y}$  均为具有  $n$  维分量的矢量,即  $\mathbf{X}, \mathbf{Y} \in \mathbb{R}^n$ ,  $\mathbf{X} = (x_1, x_2, \dots, x_n)$ ,  $\mathbf{Y} = (y_1, y_2, \dots, y_n)$ ,  $\mathbf{G}$  为需要设计的控制器.(1)式中是两个不同结构的混沌系统,设计控制器  $\mathbf{G}$  实现两个混沌系统反同步.设方程(1)的解为  $\mathbf{X}(t; t_0; \mathbf{X}_0)$  和  $\mathbf{Y}(t; t_0; \mathbf{Y}_0)$ ,且满足光滑条件.当存在  $\mathbb{R}^n$  的一个子集  $D(t_0)$ ,使初值  $\mathbf{X}_0, \mathbf{Y}_0 \in D(t_0)$ ,当  $t \rightarrow \infty$  时,若存在:

$$\xi = \lim_{t \rightarrow \infty} \|\mathbf{X}(t; t_0; \mathbf{X}_0) - \mathbf{Y}(t; t_0; \mathbf{Y}_0)\| \rightarrow 0, \quad (2)$$

则称方程(1)中响应系统与驱动系统达到反同步.

所以,实现混沌系统反同步的关键是设计一种控制器  $\mathbf{G}$ ,本文中控制器  $\mathbf{G}$  的设计采用自抗扰控制算法.

## 3 异结构混沌系统反同步的自抗扰控制

### 3.1 Lorenz 系统和 Chua 系统描述

本文以 Lorenz 系统和 Chua 系统为例,主要研究异结构混沌系统的反同步问题,将 Lorenz 系统作为驱动系统产生被跟踪信号,Chua 系统作为响应系统,两个混沌系统如(3)式和(4)式所示.

Lorenz 系统表述形式为

$$\begin{cases} \dot{x}_1 = 10(x_2 - x_1), \\ \dot{x}_2 = 28x_1 - x_2 - x_1x_3, \\ \dot{x}_3 = -\frac{8}{3}x_3 + x_1x_2. \end{cases} \quad (3)$$

Lorenz 系统初值选取  $x_1(0) = -4.47$ ,  $x_2(0) = -0.505$ ,  $x_3(0) = 28.02$  时,其混沌特性如图 1 所示.

Chua 系统表述形式为

$$\begin{cases} \dot{y}_1 = 9[y_2 - \phi(y_1)], \\ \dot{y}_2 = y_1 - y_2 + y_3 + u, \\ \dot{y}_3 = -14.286y_2, \\ y(t) = y_1(t), \end{cases} \quad (4)$$

其中,

$$\phi(y) = \frac{2}{7}y - \frac{3}{14}(|y+1| - |y-1|).$$

Chua 系统初值选取  $y_1(0) = -1.0$ ,  $y_2(0) = 0$ ,  $y_3(0) = -1.0$  时,其混沌特性如图 2 所示.

由图 1 和图 2 可以明显发现 Lorenz 系统和 Chua 系统混沌形式完全不同,因此要设计三个具有不同控制量的 Chua 系统响应系统,使其输出状态量  $y(t)$  分别与 Lorenz 系统的三个状态量  $x_1(t)$ ,  $x_2(t)$ ,  $x_3(t)$  达到反同步.

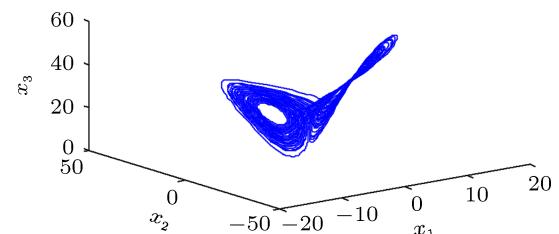


图 1 Lorenz 混沌系统混沌吸引子

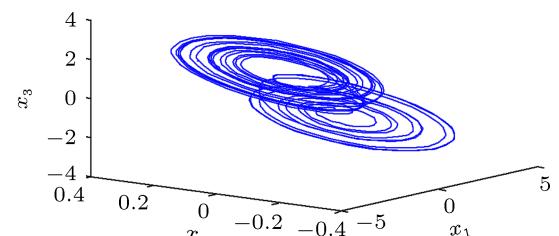


图 2 Chua 混沌系统混沌吸引子

### 3.2 Lorenz 系统和 Chua 系统反同步自抗扰控制器设计

自抗扰控制器主要由过渡安排过程(TD), ESO, NLSEF 和扰动估计补偿四部分组成.异结构混沌反同步自抗扰控制器的结构如图 3 所示.

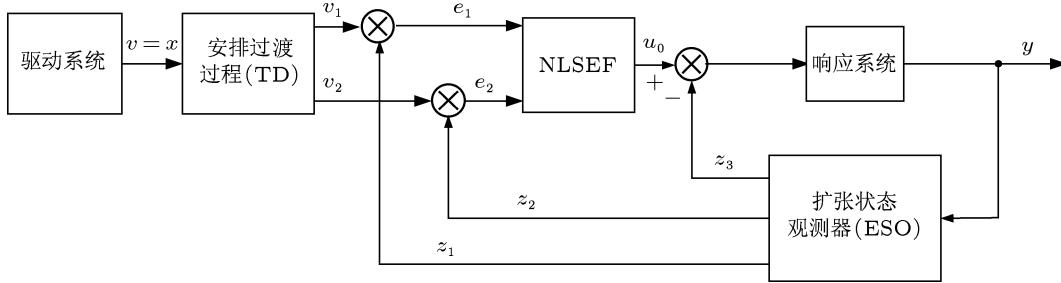


图 3 自抗扰控制器结构

本文分别针对驱动系统 Lorenz 系统的三个状态  $x_1, x_2, x_3$  设计三个自抗扰控制器, 将得到的控制量分别加到 Chua 系统上构成三个具有不同控制的响应系统, 使 Chua 系统的输出  $y(t)$  分别与 Lorenz 系统的三个状态实现反同步。以下以 Lorenz 系统  $x_1$  状态为例设计自抗扰控制器, 使 Chua 系统状态  $y$  与 Lorenz 系统状态  $x_1$  实现反同步。其余两个状态控制器设计与之类似, 只是参数要加以调整。

### 1) TD 设计

$$\begin{cases} \text{fh} = \text{fhan}(v_1 - (-v(t)), v_2, r_0, h), \\ \dot{v}_1 = v_2, \\ \dot{v}_2 = \text{fh}, \end{cases} \quad (5)$$

其中,  $v(t)$  是驱动系统需要跟踪的状态, 在这里令  $v(t) = x_1$ 。

最速综合函数  $\text{fhan}(x_1, x_2, r, h)$  的表达式如下:

$$\begin{cases} d = rh^2, \quad a_0 = hx_2, \quad y = x_1 + a_0, \\ a_1 = \sqrt{d(d+8|y|)}, \\ a_2 = a_0 + \text{sign}(y)(a_1 - d)/2, \\ s_y = (\text{sign}(y+d) - \text{sign}(y-d))/2, \\ a = (a_0 + y)s_y + a_2(1 - s_y), \\ s_a = (\text{sign}(a+d) - \text{sign}(a-d))/2, \\ \text{fhan} = -r\left(\frac{a}{d}\right)s_a - r\text{sign}(a)(1 - s_a). \end{cases}$$

### 2) ESO

$$\begin{cases} e = z_{11} - y, \\ \text{fe}_1 = \text{fal}(e, \alpha_1, \delta), \\ \text{fe}_2 = \text{fal}(e, \alpha_2, \delta), \\ \dot{z}_{11} = z_{12} - \beta_{101}e, \\ \dot{z}_{12} = z_{13} - \beta_{102}\text{fe}_1, \\ \dot{z}_{13} = -\beta_{103}\text{fe}_2, \end{cases} \quad (6)$$

$$\text{fal}(e, \alpha, \delta) = \begin{cases} \frac{e}{\delta^{\alpha-1}} & |e| \leq \delta \\ |e|^\alpha \text{sign}(e) & |e| > \delta \end{cases},$$

$\text{fal}(e, \alpha, \delta)$  函数是韩京清研究员提出的一种原点附近具有线性段的幂次函数, 它可以避免出现高频振荡现象。其中,  $e$  是误差,  $\alpha$  是指数,  $\delta$  是区分  $e$  大小的界限。

### 3) 控制量设计

本文控制器中的控制量选用  $\text{fhan}(c_{11}e_1, c_{12}e_2, r, h_1)$  非线性组合方式实现控制目标, 其形式可以表述为

$$\begin{cases} e_1 = v_1 - z_{11}, \\ e_2 = v_2 - z_{12}, \\ u_1 = \text{fhan}(c_{11}, e_1, c_{12}e_2, r, h_1) - z_{13}, \end{cases} \quad (7)$$

其中, 所选用的  $\text{fhan}(c_{11}e_1, c_{12}e_2, r, h_1)$  函数形式与 TD 中所采用的最速综合函数形式相同。

将根据状态  $x_1$  所设计的控制量施加到 Chua 系统上, 如(8)式所示。同理, 可以根据状态  $x_2$  和  $x_3$  分别设计另外两个自抗扰控制器, 生成控制量  $u_2$  和  $u_3$ 。

$$\begin{cases} \dot{y}_1 = 9[y_2 - \phi(y_1)], \\ \dot{y}_2 = y_1 - y_2 + y_3 + u_1, \\ \dot{y}_3 = -14.286y_2. \end{cases} \quad (8)$$

## 4 CPSO 的自抗扰控制器参数寻优

本文设计的自抗扰控制器所涉及的参数很多, TD 中有 2 个参数  $r_0, h$  需要整定, ESO 中有 6 个参数  $\beta_{101}, \beta_{102}, \beta_{103}, \alpha_1, \alpha_2, \delta$  需要整定, NLSEF 有 4 个参数  $c_{11}, c_{12}, r, h_1$  需要整定。这样, 整个控制系统共有 12 个参数需要整定, 将这 12 个参数协调组合使系统达到最优性能事实上是相当困难的。通过验证, ADRC 中 5 个参数  $\beta_{101}, \beta_{102}, \beta_{103}, c_{11}, c_{12}$  对控

制系统性能影响最大, 其余参数可以根据经验预先调整好并固定下来<sup>[14]</sup>, 本文选取 TD 中  $r_0 = 10000$ ,  $h = 0.001$ , ESO 中  $\alpha_1 = 0.5$ ,  $\alpha_2 = 0.25$ ,  $\delta = 0.001$ , NLSEF 中  $r = 3000$ ,  $h_1 = 0.08$ . CPSO 优化算法对 ADRC 影响最大的 5 个参数  $\beta_{101}, \beta_{102}, \beta_{103}, c_{11}, c_{12}$  进行在线实时寻优, CPSO-ADRC 系统框图如图 4 所示.

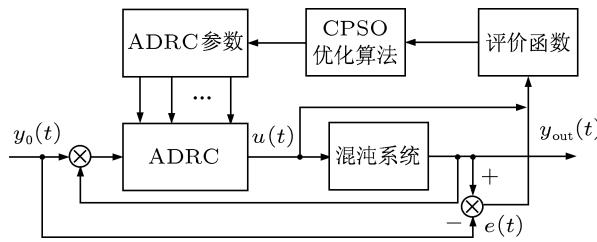


图 4 基于 CPSO 算法混沌系统自抗扰控制系统框图

## 4.1 评价函数

评价函数是用来对控制系统进行性能评价和监护的, 为获得满意的过渡过程动态特性, 使系统响应快, 运行平稳, 超调小, 本文采用 ITAE 标准, 即误差绝对值时间积分性能指标为参数选择的最小优化目标函数<sup>[15]</sup>, ITAE 标准可以描述为

$$J = \int_0^T t |e(t)| dt, \quad (9)$$

式中,  $e(t)$  为系统的输出误差,  $t$  为时间.

## 4.2 自适应惯性权重

惯性权重  $\omega$  描述了粒子上一代速度对当前代速度的影响, 控制其取值大小可调节 CPSO 算法的全局与局部寻优能力. 本文在文献 [16] 的基础上, 采用指数自适应惯性权重, 实现了 CPSO 方法增强全局搜索和局部搜索之间的平衡能力, 其表达式如 (10) 式所示:

$$\omega = (\omega_{\max} - \omega_{\min}) \times \exp \left( - \left( \tau \times \frac{t}{\text{IterMax}} \right)^2 \right) + \omega_{\min}. \quad (10)$$

由 (10) 式可知, 使用指数自适应惯性权重可以实现在算法初期有较大的惯性权重, 加强了全局搜索能力; 而在算法后期惯性权重较小, 提高了局部挖掘能力. 式中  $\tau$  根据经验选取, 一般选  $\tau \in [20, 55]$ .

## 4.3 算法流程

CPSO 流程图如图 5 所示, 具体算法如下.

**步骤 1** CPSO 算法程序初始化:

1) 初始化参数 学习因子  $c_1$  和  $c_2$ , 粒子数目 SwarmSize, 粒子维数 Dim, 速度边界  $V_{\max}$  和  $V_{\min}$ , 粒子位置边界  $X_{\max}$  和  $X_{\min}$ , 最大迭代次数 Iter<sub>max</sub> 等;

2) 粒子混沌初始化 随机产生一个  $n$  维且每个分量均在 (0, 1) 间的向量,  $z_1 = (z_{11}, \dots, z_{15})$ ; 根据 Logistic 方程  $z_{n+1} = \mu z_n (1 - z_n)$ ,  $n = 0, 1, 2, \dots$  迭代产生 (SwarmSize-1) 个向量  $z_2, \dots, z_{\text{SwarmSize}}$ , 并将  $z_i$  的各个分量映射到对应变量的取值区间, 得到 SwarmSize 个初始粒子;

3) 粒子速度初始化 根据公式  $V_{ij} = -V_{\min}(j) + 2 \times V_{\max}(j) \times \text{rand}(1)$  得到粒子的初始速度, 其中,  $V_{\min}(j)$  和  $V_{\max}(j)$  表示粒子第  $j$  个分量速度的上下界.

**步骤 2** 粒子最优位置 pibest 和粒子全局最优位置 pgbest 更新: 若粒子适应度优于个体极值 pibest, 则将新位置设置为粒子最优位置 pibest; 若粒子适应度优于全局极值 pgbest, 则将新位置设置为粒子全局最优位置 pgbest.

**步骤 3** 粒子速度和位置更新: 首先根据 (10) 式更新当代惯性权重  $\omega$ , 然后根据 (11) 和 (12) 式分别更新粒子的速度和位置

$$\begin{aligned} v_{id}(k+1) &= \omega v_{id}(k) + c_1 \text{rand}(\cdot)(\text{pibest}(k) - x_{id}(k)) \\ &\quad + c_2 \text{rand}(\cdot)(\text{pgbest}(k) - x_{id}(k)), \end{aligned} \quad (11)$$

$$x_{id}(k+1) = x_{id}(k) + v_{id}(k+1). \quad (12)$$

**步骤 4** 对最优位置 pgbset = (pgbset<sub>1</sub>, ..., pgbset<sub>Dim</sub>) 进行混沌优化:

1) 将 pgbset<sub>i</sub> ( $i = 1, 2, \dots, \text{Dim}$ ) 利用公式  $z_i = (\text{pgbset}_i - a_i) / (b_i - a_i)$  ( $i = 1, 2, \dots, \text{Dim}$ ) 映射到 Logistic 方程定义域 [0, 1] 上;

2) 用 Logistic 方程进行迭代产生混沌变量序列  $z_i^{(m)}$  ( $m = 1, 2, \dots$ );

3) 把产生的混沌变量序列通过逆映射  $\text{pgbset}_i^{(m)} = a_i + (b_i - a_i)z_i^{(m)}$  返回到原解空间, 得  $\text{pgbset}^{(m)} = (\text{pgbset}_1^{(m)}, \text{pgbset}_2^{(m)}, \dots, \text{pgbset}_D^{(m)})$  ( $m = 1, 2, \dots$ ), 其中  $[a_i, b_i]$  是第  $i$  维自变量的可行域;

4) 原解空间中, 对混沌变量经历的每个可行解  $\text{pgbset}_i^{(m)}$  ( $m = 1, 2, \dots$ ) 计算其适应值, 得到性能最好的可行解  $p^*$ .

**步骤 5** 用性能最好的可行解  $p^*$  取代当前粒子群中任意一个粒子的位置.

**步骤 6** 若满足停止条件, 则停止搜索, 输出全局最优位置, 否则返回步骤 2.

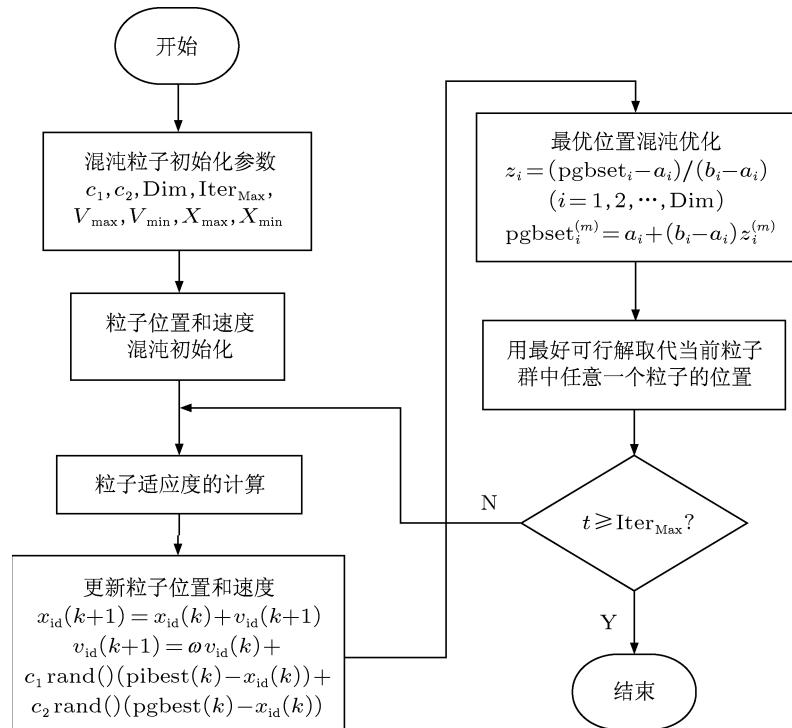


图 5 CPSO 算法流程图

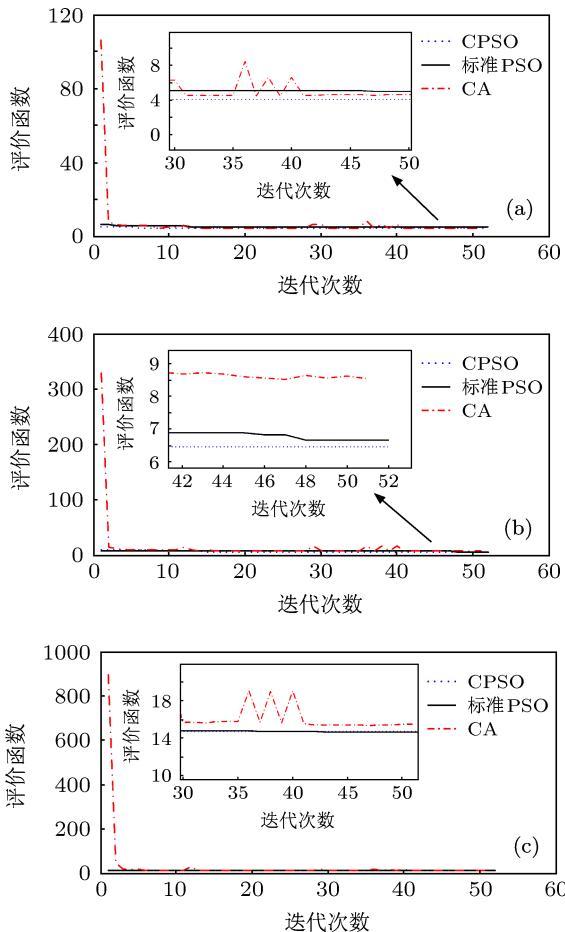


图 6 评价函数进化曲线 (a) 第一组参数寻优; (b) 第二组参数寻优; (c) 第三组参数寻优

## 5 仿真结果

### 5.1 算法比较仿真

图 6 为标准粒子群 (PSO)<sup>[17,18]</sup>, 遗传算法 (GA)<sup>[14,19]</sup>, CPSO 三种优化算法对三组 ADRC 参数整定过程中的评价函数进化曲线. 三种算法中, 种群规模均取 30 个, 迭代次数取 50 代. 由图可知, CPSO 的寻优范围要远远大于 PSO 和 GA 两种方法, 在算法初期 CPSO 算法的评价函数值迅速下降, 说明此算法具有较好的初值寻优, 加快了搜索速度, 整体提高了算法的优化效率, 并且 CPSO 可以比另两种方法找到更小的评价函数, 即更好的控制器参数.

### 5.2 异结构混沌反同步控制仿真

为了验证所设计的 CPSO-ADRC 对异结构混沌反同步控制的有效性, 利用 MATLAB 采用四阶龙格 - 库塔法, 仿真时间选为  $T = 10$  s, 驱动系统和响应系统分别选取初始状态  $x_0 = (-4.47, -0.505, 28.02)$ ,  $y_0 = (-1.0, 0.0, -1.0)$ .  $T = 5$  s 时对驱动系统各个状态上施加由 Random Number Block 产生的均值为 10 的正态分布的随机干扰信号, 如图 7 所示. 经过 CPSO 优化算法数代

优化后的三组 ADRC 参数分别为

$$\begin{aligned} & \{\beta_{101}, \beta_{102}, \beta_{103}, c_{11}, c_{12}\} \\ &= \{1958.49, 23177.31, 405.71, 75.20, 100\}, \\ & \{\beta_{201}, \beta_{202}, \beta_{203}, c_{21}, c_{22}\} \\ &= \{3098.21, 26096.40, 0, 21.19, 26.22\}, \\ & \{\beta_{301}, \beta_{302}, \beta_{303}, c_{31}, c_{32}\} \\ &= \{9891.35, 5178.09, 500, 96.81, 0.67\}. \end{aligned}$$

驱动系统和响应系统状态反同步仿真波形如图 8 所示。由图可知，两系统状态从不同初值条件出发，可以很快达到反同步。图 9 分别是系统寻优前和利用 CPSO 算法对参数寻优的状态误差曲线，可以看出经过参数寻优后系统误差明显小于系统寻优前的误差，且震荡明显减小，充分说明了 CPSO-ADRC 对异结构混沌反同步具有很好的控制效果。图 10

分别是根据驱动系统三个状态所设计控制器的控制量。此外，从上述仿真结果还可以得出，当对系统施加随机扰动后，通过控制器的调节，控制量和系统状态误差几乎没有波动，响应系统状态仍然可以迅速与驱动系统实现反同步，充分证明自抗扰控制器具有很强的抗干扰能力。

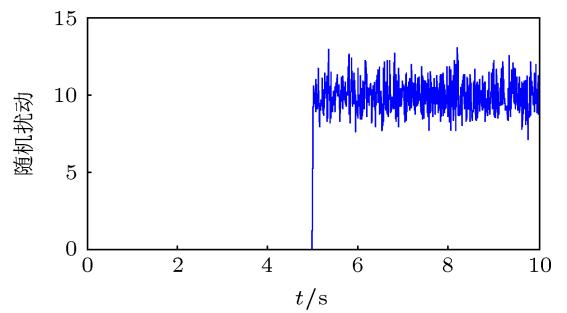


图 7 随机扰动

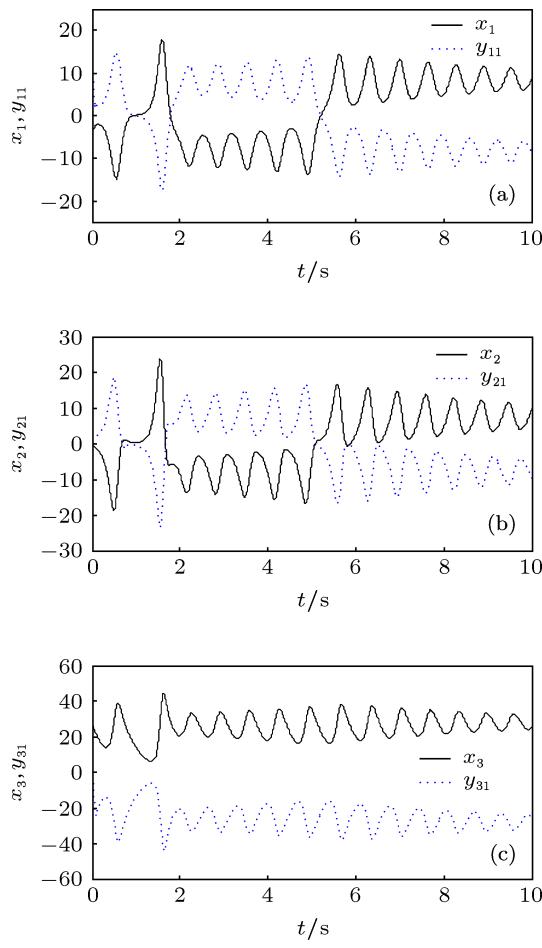


图 8 驱动系统和响应系统状态反同步仿真波形 (a) 第一个状态; (b) 第二个状态; (c) 第三个状态

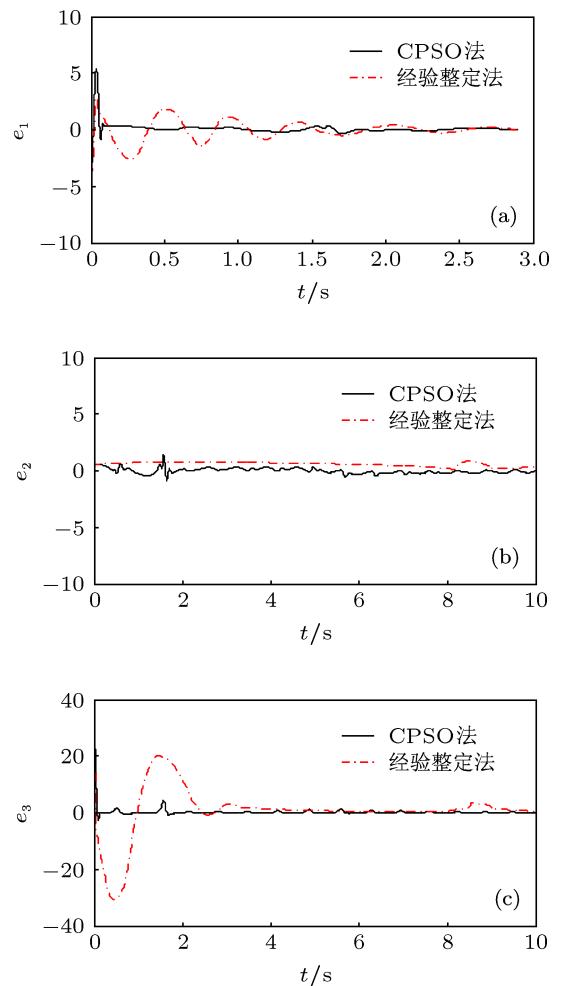


图 9 系统状态误差曲线 (a) 第一个状态; (b) 第二个状态; (c) 第三个状态

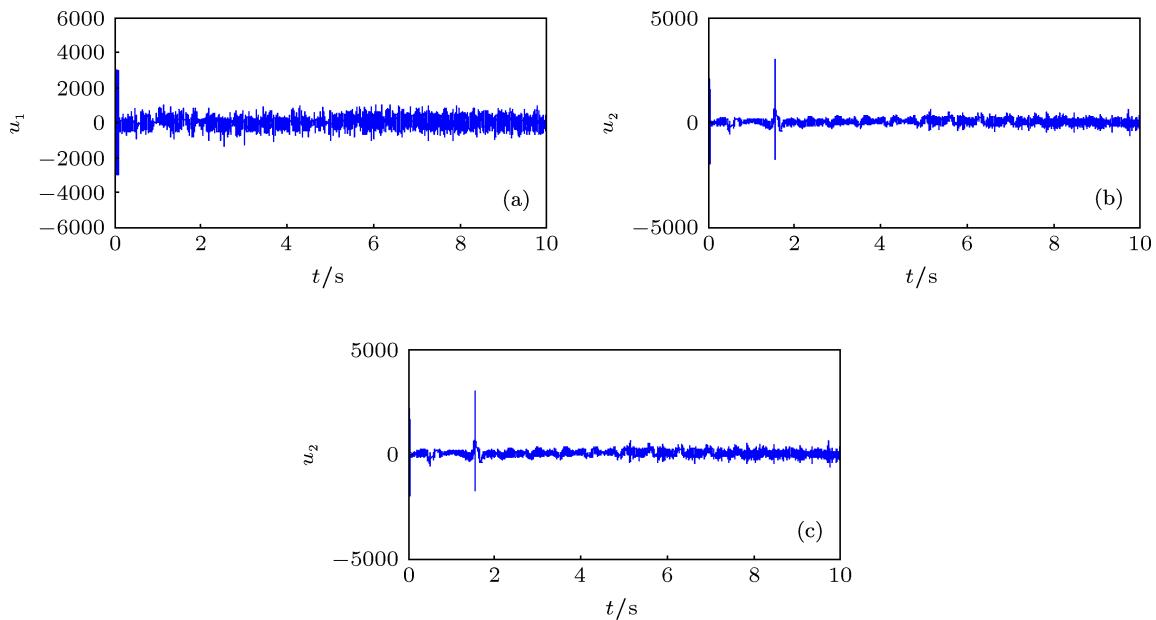


图 10 控制量 (a) 第一个控制量; (b) 第二个控制量; (c) 第三个状态控制量

## 6 结 论

本文针对两个异构混沌系统的反同步问题, 设计了自抗扰控制器, 并利用 CPSO 优化算法解决了自抗扰控制器参数多, 不利于整定和优化的问题。以 Lorenz 系统和 Chua 系统两个异结构混沌反同步控制为例进行仿真, 仿真结果表明, 经 CPSO 算

法整定参数后的 ADRC 不仅具有良好的控制效果, 使响应混沌系统与驱动混沌系统很快达到反同步, 并且当系统出现扰动时, 可以很快做出相应调整, 仍具有很好的性能。由此可见, 本文提出的 CPSO-ADRC 控制器对异结构混沌系统反同步控制具有很好的控制效果。

- [1] Liu F C, Song J Q 2008 *Acta Phys. Sin.* **57** 4729 (in Chinese) [刘福才, 宋佳秋 2008 物理学报 **57** 4729]
- [2] Liu F C, Zang X F, Song J Q 2009 *Acta Phys. Sin.* **58** 3765 (in Chinese) [刘福才, 臧秀凤, 宋佳秋 2009 物理学报 **58** 3765]
- [3] Cai N, Jing Y W, Jiang N, Zhang S Y 2009 *J. Northeastern Univ. (Natural Science)* **30** 313 (in Chinese) [蔡娜, 井元伟, 姜囡, 张嗣瀛 2009 东北大学学报(自然科学版) **30** 313]
- [4] Wang L Z, Fang H 2010 *J. Southwest Univ. National. (Natural Science Edition)* **36** 721 (in Chinese) [王兰梓, 房辉 2010 西南民族大学学报(自然科学版) **36** 721]
- [5] Wang X Y, Wu X J 2007 *Acta Phys. Sin.* **56** 1988 (in Chinese) [王兴元, 武相军 2007 物理学报 **56** 1988]
- [6] Liu F C, Li J Y, Zang X F 2011 *Acta Phys. Sin.* **60** 030504 (in Chinese) [刘福才, 李俊义, 臧秀凤 2011 物理学报 **60** 030504]
- [7] Zhu S P 2011 *Comput. Engineer. Appl.* **47** 145 (in Chinese) [朱少平 2011 计算机工程与应用 **47** 145]
- [8] Han J Q 2002 *Control Engineer.* **9** 13 (in Chinese) [韩京清 2002 控制工程 **9** 13]
- [9] Zhu L L, Yu X N, Liu L 2005 *Electron. Instrum. Custom.* **4** 64 (in Chinese) [朱丽玲, 于希宁, 刘磊 2005 仪器仪表用户 **4** 64]
- [10] Liu Z H, Zhang Y J, Zhang J, Wu J H 2011 *Acta Phys. Sin.* **60** 0195011 (in Chinese) [刘朝华, 张英杰, 张兢, 吴建辉 2011 物理学报 **60** 0195011]
- 0195011]
- [11] Wang F B, Liu J, Chen Z K, Jiao C W 2012 *Chin. J. Mech. Engineer.* **48** 32 (in Chinese) [王福斌, 刘杰, 陈至坤, 焦春旺 2012 机械工程学报 **48** 32]
- [12] Tang X L 2007 *Ph. D. Dissertation* (Chongqing: Chongqing University) [唐贤伦 2007 博士学位论文(重庆: 重庆大学)]
- [13] Wang J Z, Zhu S L, Zhao W G, Zhu W J 2011 *Expert Syst. Appl.* **38** 8151
- [14] Geng H, Yang H Y, Zhang Y A, Chen H Y 2010 *J. Syst. Simulat.* **22** 89 (in Chinese) [耿宏, 杨海云, 张永暖, 陈华媛 2010 系统仿真学报 **22** 89]
- [15] Qian M W 2012 *Small & Special Electrical Machines* **40** 40 (in Chinese) [钱苗旺 2012 微特电机 **40** 40]
- [16] An X H, Gao Y L 2008 *J. Comput. Appl.* **28** 28 (in Chinese) [安晓会, 高岳林 2008 计算机应用 **28** 28]
- [17] Hu J, Wang Z Q, Qiao S J, Gan J C 2011 *Appl. Math. Comput.* **217** 8655
- [18] Wong W K, Leung S Y S, Guo Z X 2012 *Expert Syst. Appl.* **39** 8557
- [19] Liu J K 2002 *Advanced PID Control and MATLAB Simulation* (Beijing: Publishing House of Electronics Industry) p130 (in Chinese) [刘金琨 2002 先进 PID 控制及其 MATLAB 仿真(北京: 电子工业出版社) 第 130 页]

# Anti-synchronizing different chaotic systems using active disturbance rejection controller based on the chaos particle swarm optimization algorithm

Liu Fu-Cai<sup>†</sup> Jia Ya-Fei Ren Li-Na

(Key Laboratory of Industrial Computer Control Engineering of Hebei Province, Yanshan University, Qinhuangdao 066004, China)

(Received 17 July 2012; revised manuscript received 15 February 2013)

## Abstract

For a class of continuous time different structure chaotic systems, due to the strong robustness and adaptability of active disturbance rejection controller (ADRC), we put forward a kind of the anti-synchronization of different structure chaotic systems with auto-disturbance-rejection control strategy. Because there are many parameters in ADRC and they are very difficult to set, in this paper we propose a chaotic particle swarm optimization algorithm to optimize the parameters of controller. Using Lorenz system and Chua system with different structures as an example to perform the simulation test, the results of the simulation verify that the proposed method can realize the anti-synchronization control of the different structure chaos systems faster and has a very strong anti-interference ability.

**Keywords:** anti-synchronization of different chaotic systems, active disturbance rejection controller, chaos particle swarm optimization, parameters optimization

**PACS:** 05.45.Gg, 05.45.Ac, 05.45.Pq

**DOI:** 10.7498/aps.62.120509

---

<sup>†</sup> Corresponding author. E-mail: lfc@ysu.edu.cn