

基于混合 PSO 神经网络的自整定分数阶 PID 控制器

胡海波, 黄友锐

(安徽理工大学 电气与信息工程学院, 安徽 淮南 232001)

摘 要: 提出了一种基于混合 PSO 和 RBF 神经网络的自整定分数阶 PID 控制器的设计方法. 该控制器主要由三个部分组成: (1) 分数阶 PID 控制器直接控制被控对象; (2) 利用细菌觅食算法和粒子群算法混合优化分数阶 PID 参数值, 作为初始值; (3) 利用 RBF 神经网络具有以任意精度逼近非线性函数及训练速度快的优点, 在线整定分数阶 PID 值, 并完成对被控对象的 Jacobian 信息辨识. 实验仿真结果表明: 该控制器具有响应速度快、收敛精度高、鲁棒性强等特点, 可适用于不同的对象和过程, 特别是复杂的、无确定数学模型的控制系統.

关键词: 混合 PSO; 神经网络; 自整定; 分数阶 PID

中图分类号: TP183

文献标识码: A

文章编号: 1000-7180(2010)05-0157-05

Self-tuning Fractional Order PID Based on Hybrid PSO Neural Networks

HU Hai-bo, HUANG You-rui

(College of Electrical and Information Engineering,
Anhui University of Science and Technology, Huainan 232001, China)

Abstract: A self-tuning fractional order PID controller based on hybrid PSO and neural networks is presented. It consists of three parts. In the first part, a fractional order PID controller directly controls the object. In the second part, a group of fractional order PID parameters are obtained by hybrid PSO. The third part, RBF neural networks is used to optimize and adjust the fractional order PID parameters on-line by exploiting the nonlinear mapping capabilities of neural networks and training fast and get Jacobian information for the object. The simulation results show that the controller has a fast response speed, high convergence and robustness. It can be used to control different objects and processes, specially complex, non-determination mathematical model control system.

Key words: hybrid PSO; neural networks; self-tuning; fractional order PID

1 引言

在一些复杂的被控系统中, 传统的 PID 控制器很难达到精确的控制效果. 1999 年提出的分数阶 PID 控制器^[1] (简称 $PI^{\lambda}D^{\mu}$) 越来越受到关注. 因为 $PI^{\lambda}D^{\mu}$ 控制器不仅有传统 PID 控制器的 K_p, K_i, K_d 这三个参数, 还有积分阶次 λ 和微分阶次 μ 两个可调参数, 总共五个参数的非线性关系使得参数的调节范围扩大、难度增大, 却能够更灵活、精确地控制受控对象. 对整数 PID 参数的优化已有许多很好的方法^[2-3], 而对分数阶 PID 控制器的整定, 也出现了

新的方法^[4-5], 但是这都是在已知整定模型的情况下通过实验的方法调节出最佳参数值.

对于复杂的实际系统或在没有确定的数学模型的情况下, 文中将细菌觅食算法和粒子群算法的混合优化和神经网络同 $PI^{\lambda}D^{\mu}$ 有机的联系起来, 利用细菌觅食算法和粒子群算法的混合优化具有全局搜索能力和收敛速度快、收敛精度高的特点, 对 $PI^{\lambda}D^{\mu}$ 控制器的 5 个参数值进行优化, 作为初始值; 神经网络具有逼近任意非线性函数的能力, 径向基函数网络的优势在于其是一种局部逼近网络, 因此训练速度快, 通过对 $PI^{\lambda}D^{\mu}$ 参数值的在线调整, 并对控制系

统的 Jacobian 信息进行辨识,实现具有最优组合的 $PI^{\lambda}D^{\mu}$ 控制.被控对象在其控制下显示出很好的动态响应和收敛速度快、精度高的特点.

2 基于混合 PSO 与 RBF 神经网络的自整定分数阶 PID 控制原理

在 $PI^{\lambda}D^{\mu}$ 控制中, $[K_p, K_i, K_d, \lambda, \mu]$ 5 个参数值的整定优化非常重要,这五种控制作用在形成控制量中相互制约又相互配合,在其无穷变化的非线性关系中寻找出最优的组合以取得最好的控制效果.系统初始参数值的选取对神经网络的学习效率有很大影响,细菌觅食算法和粒子群算法的混合优化算法具有全局搜索能力强、收敛速度快的特点,则在自整定 $PI^{\lambda}D^{\mu}$ 控制前先用混合粒子群算法积累有关 $PI^{\lambda}D^{\mu}$ 参数的先验知识,解出的 $PI^{\lambda}D^{\mu}$ 参数值再传递给神经网络学习,最后得出最优的 $[K_p, K_i, K_d, \lambda, \mu]$ 参数值. RBF 神经网络在线整定 $PI^{\lambda}D^{\mu}$ 参数值,并对被控对象的 Jacobian 信息辨识,提高被控对象的输出对控制输入的灵敏度.基于混合 PSO 神经网络的自整定分数阶 PID 控制器的结构如图 1 所示.

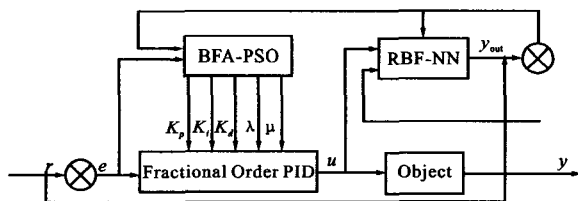


图1 自整定 $PI^{\lambda}D^{\mu}$ 控制器结构图

该控制器由三个部分组成:(1) $PI^{\lambda}D^{\mu}$ 控制器:直接闭环控制被控对象,在线调整 $[K_p, K_i, K_d, \lambda, \mu]$ 5 个参数值;(2) 细菌粒子群混合优化算法:优化整定 $PI^{\lambda}D^{\mu}$ 控制参数的初始值;(3) RBF 神经网络:对被控对象的 Jacobian 信息辨识,动态的观测对象的输出对控制输入的灵敏度信息.

3 分数阶 PID 控制器

分数阶微积分是微分和积分的阶次是任意阶的理论,随着其在控制中的研究,出现了控制理论的新领域,即分数阶控制理论与分数阶控制器.分数阶控制系统是基于分数阶微积分控制理论建立起来的.以分数阶控制理论的角度,闭环控制系统有 4 类:整数阶对象模型与整数阶控制器,整数阶对象模型与分数阶控制器,分数阶对象模型与整数阶控制器,分数阶对象模型与分数阶控制器.基于分数阶理论提

出的分数阶控制器主要有三种^[6]: TID 控制器、 $PI^{\lambda}D^{\mu}$ 控制器、分数阶滞后补偿器.前两种控制器直接与分数阶微积分相关,第三种是由经典的滞后补偿器扩展而来.分数阶 PID 控制器是整数 PID 控制器的广义表达形式,将整数阶的积分项 λ 和微分项 μ 的阶次扩展到了分数领域,其中积分阶次 λ 和微分阶次 μ 可以是任意的实数.它的时域表达式为:

$$u(t) = K_p e(t) + K_i D_t^{-\lambda} e(t) + K_d D_t^{\mu} e(t) \quad (1)$$

其中, $D_t^{\alpha} \equiv {}_a^C D_t^{\alpha}$ 为 Caputo 定义, $\lambda > 0$ 为积分阶次, $\mu > 0$ 为微分阶次. K_p, K_i, K_d 是控制器的控制参数.

实际上,在分数阶微积分的定义下,分数阶 $PI^{\lambda}D^{\mu}$ 控制器是一个无限维数的滤波器.其设计原理是根据系统的控制性能指标和闭环特征方程的零点来设计和优化控制参数的.

对式(1)求拉普拉斯变化为:

$$L\{{}_a^C D_t^{\alpha} f(t)\} = S^{\alpha} F(s) - \sum_{k=0}^{N-1} S^{\alpha-k-1} f^{(k)}(0)$$

得到分数阶 PID 控制器的传递函数为:

$$G(s) = K_p + K_i S^{-\lambda} + K_d S^{\mu}$$

当 $\lambda = 1$ 和 $\mu = 1$ 就是整数阶 PID 控制器;当 $\lambda = 0$ 时为 PD^{μ} 控制器;当 $\mu = 0$ 时为 PI^{λ} 控制器;当 $K_i = 0, \mu = 1$ 时为 PD 控制器;当 $K_d = 0, \lambda = 1$ 时为 PI 控制器.可见,以上的任一种控制器都是分数阶 PID 控制器的一种特殊情况.通过合理的整定参数,分数阶 PID 控制器能更好地、更精确地提高系统的控制效果.

$PI^{\lambda}D^{\mu}$ 控制器的每项都有明确的控制意义, P 是比例系数, I^{λ} 是误差的 λ 阶次的积分, D^{μ} 是误差的 μ 阶次的微分.

4 基于 BFA-PSO 的分数阶 PID 参数的优化整定

4.1 混合粒子群优化算法

4.1.1 粒子群优化算法

粒子群优化算法(PSO)在求解优化问题时,每个问题的解被看作一个个的微粒,所有粒子都有一个被优化的函数决定其适应度值,且还有一个速度决定粒子飞行的方向和距离,该飞行速度根据个体的飞行经验和群体的飞行经验进行动态的调整.在每一次迭代中,粒子通过跟踪两个“极值”来更新自己:一个是个体极值 P_{id} ,即粒子自身目前所找到的最优解;另一个是全局极值 P_{gd} ,即整个种群目前找

到的最优解.粒子的速度和位置的更新公式如下:

$$V_{id}^t = \omega \cdot V_{id}^{t-1} + c_1 \cdot \text{rand}_1() \cdot (P_{id}^{t-1} - X_{id}^{t-1}) + c_2 \cdot \text{rand}_2() \cdot (P_{gd}^{t-1} - X_{id}^{t-1}) \quad (2)$$

$$X_{id}^{t+1} = X_{id}^t + V_{id}^t \quad (3)$$

式中, X 代表粒子当前位置; V 代表粒子的速度; rand 是产生 $(0,1)$ 之间随机数的函数; ω 为惯性权重系数,它使粒子保持运动特性,使其有扩展搜索空间的趋势,有能力探索新的区域.较大的惯性权重 ω 值有利于跳出局部极小点,较小的惯性权重 ω 值有利于算法收敛,因此就有了自适应调整 ω 的策略的提出,即随着迭代的进行,线性减小 ω 的值.对全局搜索,通常希望在前期有较高的搜索能力,而在后期有较高的开发能力以加快收敛速度.通常采用下式来调整:

$$\omega = \omega - \text{iter}(\omega_{\max} - \omega_{\min})/\text{iter}_{\max} \quad (4)$$

式中, iter 指当前的代数, $\text{iter}_{\max}$ 指运行的最大迭代次数, $\omega_{\max}$ 、 $\omega_{\min}$ 是惯性权重系数的最大值和最小值,加速常数 C_1 和 C_2 代表将每个粒子推向 P_{id} 和 P_{gd} 位置的统计加速项的权重,一般取 $(0,2)$ 之间的随机数^[7].

4.1.2 细菌觅食算法

细菌觅食算法 (Bacterial Foraging Algorithm, BFA)^[8]是由 K. M. Passino 教授于 2002 年提出的一种新型仿生类算法,该算法基于 E. coli 大肠杆菌在人体肠道内吞噬事物的行为.大肠杆菌在觅食过程中,通过身体的鞭毛顺时针(逆时针)的转动来推动自身的运行或翻转,并以 E/T (E 是细菌群得到的能量, T 是搜寻的时间)的速率一步步地寻找营养最丰富的地方.细菌群的觅食行为可以看作是不断优化过程,优化问题的解对应搜索空间内细菌的状态,即优化函数适应度值.在整个过程中,细菌的生命周期会结束或者被杀死,或者在不同的温度下繁殖. BFA 包括趋化 (chemotactic)、繁殖 (reproduction)、趋散 (elimination - dispersal) 三个步骤.

(1) 趋化是细菌朝含有营养的区域聚集的行为.在趋化过程中,细菌运动模式包括翻转 (tumble) 和前进 (swim). 翻转是细菌向任意方向移动单位步长,其更新公式为:

$$x(j+1, k, l) = x(j, k, l) + c(i)q(j) \quad (5)$$

式中, $c(i) > 0$ 是细菌在任意方向被指定的步长的大小.细菌完成一次翻转,检查适应度值是否改变,若适应度值得到改善,细菌将沿同一方向继续移动若干步,如此循环直至适应度值不再改善,或达到设定的移动步数临界值,此过程定义为前进.

(2) 当满足设定的趋化次数,细菌将进行分裂繁殖.细菌的分裂繁殖过程遵循自然界“优胜劣汰,适者生存”原则.适应度值好(即细菌终生觅食过程中消耗小)的细菌分裂成两个子细菌,子细菌继承了母细菌的特性,拥有和母细菌相同的位置和步长.

(3) 适应度值差(即细菌终生觅食过程中消耗大)或者生命周期结束的细菌将被趋散.为了确保细菌总数的不变,将趋散相同数目的细菌,在每次趋散中,细菌或者死亡或者以一定概率被趋散到空间里的任意位置.

$$P\{j, k, l\} = \{X^i(j, k, l) \mid i = 1, 2, \dots, s\} \quad (6)$$

式(6)说明 s 个细菌在第 j 个趋化步,第 k 个繁殖步和第 l 个趋散步中第 i 个细菌的位置.

4.1.3 混合粒子群算法

在解问题过程中,粒子群在解空间内不断迭代,如果某粒子发现某个当前位置为最优位置时,其他粒子将迅速向其靠拢,如果该最优位置为局部最优,粒子群就无法在解空间内重新搜索,使粒子群聚集在局部最优处而陷入“早熟”中.为了改进这种缺点,文中将 PSO 和具有趋化、繁殖和趋散特点的细菌觅食算法相结合,根据下式来混合优化:

$$J(i, j, R, ED) = F(P(:, i, j, R, ED)) \quad (7)$$

式中, i, j 分别代表细菌(也是粒子)的数量和趋化变量; R 是细菌的繁殖变量; ED 是细菌的驱散变量; F 函数是适应度函数.

细菌觅食算法特有的趋化过程可以加强细菌的局部搜索能力,繁殖过程可以加快细菌的搜索速度,趋散过程是细菌群搜索到全局最优位置的保证.

在解空间内,细菌的更新是由趋化、繁殖和驱散决定,相应的粒子也会根据细菌的更新和适应度函数来更新自己的位置和速度,根据式(8)、(9)不断更新个体极值和全局极值,直到迭代结束.

$$\text{plocal_position}(:, i, j) = \text{current_position}(:, j, \text{index}(i, :)) \quad (8)$$

$$\text{global_position} = \text{current_position}(:, I, \text{index}(I, :)) \quad (9)$$

基于以上可以看出,具有全局搜索能力的细菌觅食算法根据细菌的更新来带动粒子群的更新,原本因为根据适应度函数更新而易陷入局部收敛中的 PSO,现多了细菌的趋化、繁殖和驱散操作来更新自己的位置和速度,弥补了粒子群算法易陷入局部最优值的缺陷,从而求解出全局最优值.

4.2 BFA-PSO 优化 $PI^{\lambda}D^{\mu}$ 参数初始值

对于 $PI^{\lambda}D^{\mu}$ 控制器 BFA-PSO 采用的是实数编码, 参数寻优中的粒子可以直接编码为 $[K_p, K_i, K_d, \lambda, \mu]$, 采用误差绝对时间积分性能指标为参数选择的最小目标函数. 为了防止控制能量过大, 在目标函数中加入控制输入的平方项, 即:

$$J = \int_0^{\infty} (\omega_1 |e(t)| + \omega_2 u^2(t)) dt + \omega_3 t_s$$

其中 $e(t)$ 为控制误差, $u(t)$ 为控制器输出, $\omega_1, \omega_2, \omega_3$ 为权值, t_s 为上升时间.

为了避免系统超调, 采用惩罚功能, 即一旦产生超调, 超调量将作为最优指标的一项, 即:

$$\text{If } ey(t) < 0, J = \int_0^{\infty} (\omega_1 |e(t)| + \omega_2 u^2(t) + \omega_4 |ey(t)|) dt + \omega_3 t_s$$

其中, $\omega_4 \gg \omega_1$, $ey(t) = y(t) - e(t-1)$, $y(t)$ 为被控对象输出^[9].

BFA-PSO 实现步骤如下:

Step 1: 初始化. 设定粒子群总数为 60, 粒子大小为 5 ($[K_p, K_i, K_d, \lambda, \mu]$ 5 个参数), C_1 和 C_2 取值 2, 惯性系数 $\omega_{\max} = 0.9$, $\omega_{\min} = 0.4$, $\omega_1 = 0.999$, $\omega_2 = 0.001$, $\omega_3 = 2.0$, $\omega_4 = 100$, 采样时间为 0.001s, 最大迭代次数为 60 次.

Step 2: 初始化. 设定细菌数目为 40, 趋化步长为 20, 繁殖步长为 4, 趋散步长为 2, 细菌每代繁殖数目为 20, 驱散概率为 0.25.

Step 3: 将每个粒子(细菌)的位置向量依次作为 $PI^{\lambda}D^{\mu}$ 控制器参数, 并计算粒子群的适应度值, 初始化 P_i, P_g .

Step 4: 细菌翻转, 搜寻最优位置, 更新粒子群的适应度值, P_i, P_g , 位置.

Step 5: 判断粒子群的适应度值是否得到改善, 是则继续进行下一步, 否则转到 Step4.

Step 6: 细菌是否达到趋化步数, 是则细菌繁殖和趋散, 否则转到 Step4.

Step 7: 终生消耗最小的细菌分裂, 消耗多的细菌以一定概率被趋散, 产生新的细菌群, 转到 Step4.

Step 8: 判断是否满足最大迭代次数, 是则结束, 否则转到 Step4.

5 RBF 辨识网络对 $PI^{\lambda}D^{\mu}$ 的参数调整

径向基函数(radial basis function, RBF)^[9]方法是在高维空间进行插值的一种技术. Broomhead 和 Love 在 1998 年率先使用该技术, 提出了神经网络

学习的一种新手段. RBF 网络由三层组成, 输入层节点只是传递输入信号到隐层, 隐层节点(也称 RBF 节点)由像高斯核函数那样的辐射状作用函数构成, 而输出层节点通常是简单的线性函数. 隐层节点的核函数对靠近核函数的中央范围的输入信号产生局部响应, 隐层节点将产生较大的输出. 由此可看出 RBF 网络具有局部逼近能力, 从而大大加快了学习速度.

仿真实验中控制误差为:

$$e(k) = u(k) - y(k)$$

RBF 辨识网络的三个输入向量为:

$$X = [e(k) - e(k-1), e(k), e(k) - 2e(k-1) + e(k-2)], \text{隐层个数为 6, 如图 2 所示.}$$

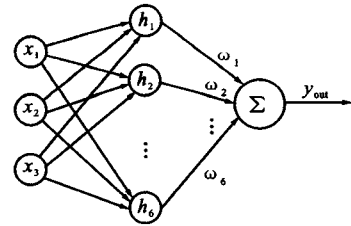


图 2 RBF 网络结构图

RBF 网络的径向基向量 $H = [h_1, h_2, h_3, h_4, h_5, h_6]^T$, 高斯基函数 $h_j = \exp(-\frac{\|X - C_j\|^2}{2b_j^2})$, $j = 1, 2, \dots, 6$. 网络的第 j 个节点的中心矢量 $C_j = [C_{j1}, C_{j2}, C_{j3}]^T$, $i = 1, 2, 3$. $b_j > 0$ 为网络的基宽度参数. 网络的权向量为 $w = [w_1, w_2, \dots, w_6]$, RBF 辨识网络的输出为 $y_{out} = w_1 h_1 + w_2 h_2 + \dots + w_6 h_6$, 性能指标函数为 $J' = \frac{1}{2} (y(k) - y_{out}(k))^2$.

由梯度下降法得输出权、节点中心和节点基宽参数的迭代算法如下:

$$w_j(k) = w_j(k-1) + \eta (y(k) - y_{out}(k)) h_j + \alpha (w_j(k-1) - w_j(k-2))$$

$$\Delta b_j = (y(k) - y_{out}(k)) w_j h_j \frac{\|X - C_j\|^2}{b_j^3}$$

$$b_j(k) = b_j(k-1) + \eta \Delta b_j + \alpha (b_j(k-1) - b_j(k-2))$$

$$\Delta c_{ji} = (y(k) - y_{out}(k)) w_j \frac{x_{ji} - c_{ji}}{b_j^2}$$

$$c_{ji}(k) = c_{ji}(k-1) + \eta \Delta c_{ji} + \alpha (c_{ji}(k-1) - c_{ji}(k-2))$$

对于 $PI^{\lambda}D^{\mu}$ 控制器, RBF 神经网络整定指标为:

$$E(k) = \frac{1}{2} e(k)^2$$

通过梯度下降法, $[K_p, K_i, K_d, \lambda, \mu]$ 的调整如下:

下:

$$\Delta k_p = -\eta \frac{\partial E}{\partial k_p} = \eta e(k) \frac{\partial y}{\partial u} x_1$$

$$\Delta k_i = -\eta \frac{\partial E}{\partial k_i} = \eta e(k) \frac{\partial y}{\partial u} x_2$$

$$\Delta k_d = -\eta \frac{\partial E}{\partial k_d} = \eta e(k) \frac{\partial y}{\partial u} x_3$$

$$\Delta \lambda = -\eta \frac{\partial E}{\partial \lambda} = \eta e(k) \frac{\partial y}{\partial u} (-\lambda x_2^{1-\lambda})$$

$$\Delta \mu = -\eta \frac{\partial E}{\partial \mu} = \eta e(k) \frac{\partial y}{\partial u} \mu x_2^{\mu-1}$$

式中, η 为学习速率, α 为动量因子, $\frac{\partial y}{\partial u}$ 为被控对象的 Jacobian 信息, 由 RBF 辨识而得.

6 实验仿真

为了验证该整定方法的有效性, 取被控对象为:

$y(k+1) = g[y(k), u(k)]$, 未知函数为:

$$g(x_1, x_2) = \frac{-0.1x_1 + x_2}{1 + x_1^2}$$

假设辨识模型为:

$$\hat{y}(k+1) = \hat{f}[y(k), u(k)],$$

$$y(0) = 0, u(0) = 0.$$

先利用 BFA-PSO 整定优化 $PI^\lambda D^\mu$ 的初始参数值, 如图 3、图 4 所示.

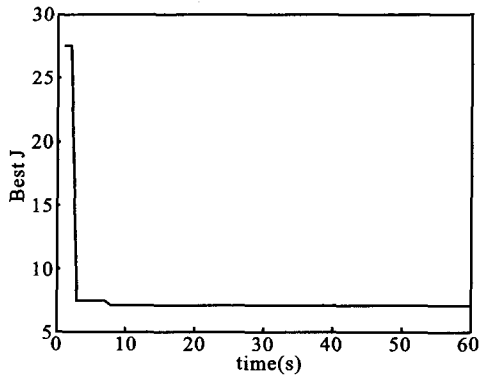


图3 BFA-PSO 优化 $PI^\lambda D^\mu$ 收敛曲线图

解出初始值为: $k_p = 0.10089, k_i = 0.8795, k_d = 0.01467, \lambda = -0.014, \mu = 1.009$.

在调试环境、初始条件、循环次数相同的条件下, 用径向基网络优化分数阶 PID 参数值 (RBF-fractional order PID), 并和文中提出的整定方法相比较, 可以明显看出 BFA-PSO/RBF 优化后的 $PI^\lambda D^\mu$ 具有响应速度快、收敛精度高的特点, 如图 5 所示.

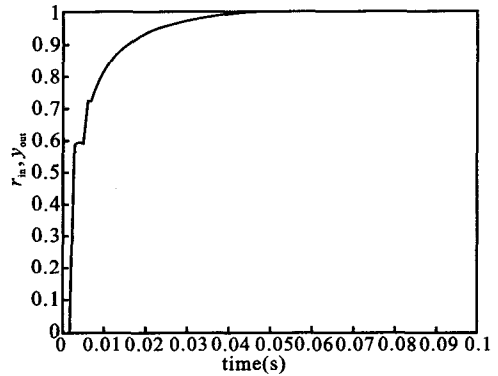


图4 BFA-PSO 优化 $PI^\lambda D^\mu$ 动态响应图

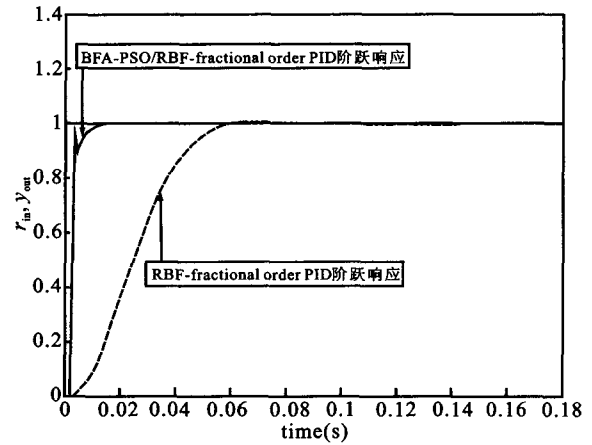


图5 $PI^\lambda D^\mu$ 控制器阶跃响应比较

7 结束语

文中提出了一种基于细菌觅食算法和粒子群优化混合算法和 RBF 神经网络的自整定分数阶 PID 控制器的设计方法, 实现了控制器参数的优化整定. 先利用混合粒子群算法得出控制器参数的初始值, 在通过 RBF 神经网络在线调整参数值, 得到最优值, 并对被控对象实现灵敏度辨识, 提高系统的精确度. 从实验仿真结果可知: 基于混合 PSO 神经网络的自整定分数阶 PID 控制器有很好的控制效果, 更适用于复杂的、实时性高的实际系统.

参考文献:

- [1] 李明河, 王萌, 施艳艳. 基于动态神经网络在线辨识的 PID 控制[J]. 微电子学与计算机, 2007, 19(16): 43-46.
- [2] 黄友锐. 基于遗传神经网络的自整定 PID 控制器[J]. 系统仿真学报, 2003, 15(11): 1628-1630.
- [3] 孙惠杰. 转阀式直接驱动伺服阀的参数自调整模糊 PID 控制[J]. 微电子学与计算机, 2008, 25(9): 141-143.

(下转第 166 页)

以,可以用 TCED 算法取代 TALK 算法.另外,TCED 算法的不足是随着时间片长度 Δt 减小到一定程度时,唤醒次数会大量增加,能量消耗反而会增加,所以 TCED 算法在 $\Delta t \leq 20\text{ms}$ 时通常是不适用的,有关这种情况的算法优化还有待研究.

参考文献:

- [1] 徐懿,李丽,高明伦,等. 纳米集成电路静态功耗机理及低功耗设计技术[J]. 微电子学与计算机, 2007, 24(5): 184 - 188.
- [2] 张孝坤,王继安,徐赏林,等. 一种可用于模块化设计的热关断电路[J]. 微电子学与计算机, 2007, 24(6): 130 - 136.
- [3] Berkeley BSIM3 device models. URL[EB/OL]. [2009 - 06 - 10]. <http://www-device.EECS.Berkeley.edu/bsim3/>.
- [4] Lin Yuan, Sean Leventhal, Gang Qu. Temperature -

aware leakage minimization technique for real - time systems[J]. Computer - Aided Design, 2006(6): 761 - 764.

- [5] Skadron K, Abdelzaher T, Stan M R. Control - theoretic techniques and thermal - RC modeling for accurate and localized dynamic thermal management[J]. High - Performance Computer Architecture, 2002(2): 17 - 28.
- [6] He L, Liao W, Stan M R. System level leakage reduction considering the interdependence of temperature and leakage[J]. Design Automation Conference, 2004(41): 12 - 17.
- [7] Skadron K. Temperature - aware microarchitecture: modeling and implementation[J]. ACM Trans on Architecture and Code Optimization, 2004, 1(1): 94 - 125.

作者简介:

张昌明 男, (1987 -). 研究方向为低功耗设计.

骆祖莹 男, (1968 -), 博士, 副教授. 研究方向为低功耗 IC 设计和 IC 物理设计.

(上接第 161 页)

- [4] Hongsheng Li, Yangquan Chen. A fractional order proportional and derivative (FOPD) controller tuning algorithm [C]// Chinese Control and Decision Conference. China: Yantai, 2008: 4059 - 4063.
- [5] Serdar E Hamamci. Stabilization using fractional - order PI and PID controllers[J]. Nonlinear Dyn, 2008(51): 329 - 343.
- [6] Dingyu Xue, Yangquan Chen. A comparative introduction of four fractional order controller[C]// Congress on Intelligent Control and Automation. China, Shanghai, 2002.
- [7] 高尚, 杨静宇. 群智能算法及其应用[M]. 北京: 中国水利水电出版社, 2007.

- [8] Passio K M. Biomimicry of social foraging bacteria for distributed optimization: models, principles, and emergent behaviors[J]. Journal of Optimization Theory and Applications, 2002; 115(3): 603 - 628.

- [9] 周开利, 康耀红. 神经网络模型及其 MATLAB 仿真程序设计[M]. 北京: 清华大学出版社, 2007, 11.

作者简介:

胡海波 女, (1985 -), 硕士研究生. 研究方向为控制理论与控制工程.

黄友锐 男, (1971 -), 教授, 硕士生导师. 研究方向为计算机控制和智能信息处理.