

基于粒子群优化算法的分数阶 PID 控制器设计

赵华东, 宋保业, 张建胜, 许琳

(山东科技大学 电气与自动化工程学院, 山东 青岛 266590)

摘要: 分数阶 PID 控制器具有可变的微分和积分阶次, 通过调整控制器参数可以获得更好的控制性能。本文基于粒子群优化算法设计分数阶 PID 控制器。首先介绍分数阶 PID 和粒子群优化算法, 然后给出分数阶 PID 控制系统结构、分数阶微积分算子的近似算法和分数阶 PID 控制器设计的仿真流程, 最后通过 MATLAB/Simulink 对算例进行控制器设计仿真。仿真结果表明, 通过粒子群寻优能够获得满意的分数阶 PID 控制器参数, 满足对控制性能的要求。

关键词: 粒子群优化; 分数阶控制器; 比例-积分-微分

中图分类号: TP273

文献标志码: A

文章编号: 1672-3767(2017)04-0060-06

DOI: 10.16452/j.cnki.sdkjzk.2017.04.009

Fractional-order PID Controller Design Based on PSO Algorithm

ZHAO Huadong, SONG Baoye, ZHANG Jiansheng, XU Lin

(College of Electrical Engineering and Automation, Shandong University of
Science and Technology, Qingdao, Shandong 266590, China)

Abstract: The performances of the fractional-order PID controller could be advanced by adjusting two of the variable parameters, i. e. differential order and integral order. This paper is concerned with the design of fractional-order PID controller based on particle swarm optimization. Firstly, the fractional-order PID controller and particle swarm optimization were introduced respectively. Then the structure of the fractional-order PID controller, the approximation algorithm of the fractional-order differ-integral operator, and the simulation flowchart of the fractional-order PID controller design were presented. In the end, the controller design was simulated for the examples by MATLAB/Simulink. The simulation results show that with particle swarm optimization, the satisfying fractional-order PID controller parameter could be obtained to achieve better control performances.

Key words: particle swarm optimization; fractional-order controller; PID(proportional-integral-derivative)

PID(比例-积分-微分, proportional-integral-derivative)控制器是目前工业生产中应用最广泛的控制器, 而近年来发展起来的分数阶 PID 控制器在经典 PID 控制器基础上增加了两个可调参数, 从而可以在原有 PID 控制器基础上进一步优化控制性能、提高控制品质, 其适用对象包括分数阶系统和整数阶系统^[1]。

近年来, 大量的启发式智能优化算法被用于 PID 控制器参数整定和分数阶 PID 控制器设计, 如文献[2]提出基于改进遗传算法的分数阶 PID 控制器, 显著优化系统性能并提高了系统鲁棒性。文献[3]提出一种基于改进蜂群算法的分数阶 PID 控制器设计方法, 新算法能够扩大搜索范围避免陷入局部极小点, 从而获

收稿日期: 2017-03-22

基金项目: 山东省高等学校科技计划项目(J14LN34)

作者简介: 赵华东(1991—), 男, 山东泰安人, 硕士研究生, 主要从事分数阶控制理论与应用的研究。

宋保业(1982—), 男, 山东青岛人, 博士, 讲师, 主要从事智能优化算法、移动机器人控制的研究, 本文通信作者。
E-mail: songbaoye@gmail.com

得最优的控制器参数。此外,禁忌搜索算法^[4]、差分进化算法^[5]、粒子群优化算法^[6]等也被用于分数阶 PID 控制器设计。

本文基于粒子群优化 (particle swarm optimization, PSO) 算法给出一种分数阶 PID 控制器设计的 MATLAB/Simulink 仿真方法。首先介绍分数阶 PID 控制器和粒子群优化算法,然后给出基于粒子群优化算法的分数阶 PID 控制器设计方法和仿真流程,最后通过算例比较验证设计结果的优越性。

1 分数阶 PID 控制器

分数阶微积分是经典微积分的一般化,其微积分阶次从整数扩展到非整数,包括任意实数甚至复数^[7]。分数阶微积分的定义有多种形式,其中 Grunwald-Letnikov (GL) 分数阶微积分定义和 Riemann-Liouville (RL) 分数阶微积分定义是最常用的两种,并且可以证明在实际物理系统和工程应用中,两者是完全等效的^[8]。对连续可导函数 $f(t)$ 的 r 阶 Grunwald-Letnikov 微积分定义如下^[9]:

$${}_a D_t^r f(t) = \lim_{h \rightarrow 0} h^{-r} \sum_{j=0}^{\lceil \frac{t-a}{h} \rceil} (-1)^j \binom{r}{j} f(t-jh), \quad (1)$$

其中, $\lceil \cdot \rceil$ 表示整数部分, ${}_a D_t^r$ 为分数阶微积分算子, r 为其阶数 (r 为正表示微分, r 为负表示积分), a 和 t 分别为积分上下限。Riemann-Liouville 微积分定义如下^[9]:

$${}_a D_t^r f(t) = \frac{1}{\Gamma(n-r)} \frac{d^n}{dt^n} \int_a^t \frac{f(\tau)}{(t-\tau)^{r-n+1}} d\tau, \quad (2)$$

其中, $n-1 < r < n$, $\Gamma(\cdot)$ 为伽马函数。根据以上分数阶微积分定义并考虑零初始条件,分数阶微积分的拉普拉斯变换为:

$$L\{{}_a D_t^r f(t)\} = s^r F(s). \quad (3)$$

其中, $L\{\cdot\}$ 表示拉普拉斯变换, s 为拉普拉斯算子。

分数阶动态系统可用如下的分数阶微分方程描述:

$$\begin{aligned} & a_n D^{\alpha_n} y(t) + a_{n-1} D^{\alpha_{n-1}} y(t) + \dots + a_0 D^{\alpha_0} y(t) \\ & = b_m D^{\beta_m} u(t) + b_{m-1} D^{\beta_{m-1}} u(t) + \dots + b_0 D^{\beta_0} u(t), \end{aligned} \quad (4)$$

其中, $D^r \equiv {}_a D_t^r$, $a_k (k=0, \dots, n)$ 和 $b_k (k=0, \dots, m)$ 为常数, $\alpha_k (k=0, \dots, n)$ 和 $\beta_k (k=0, \dots, m)$ 为任意实数。不失一般性,可假设 $\alpha_n > \alpha_{n-1} > \dots > \alpha_0$ 及 $\beta_m > \beta_{m-1} > \dots > \beta_0$ 。考虑(3)式,则分数阶动态系统的传递函数为:

$$G(s) = \frac{b_m s^{\beta_m} + b_{m-1} s^{\beta_{m-1}} + \dots + b_0 s^{\beta_0}}{a_n s^{\alpha_n} + a_{n-1} s^{\alpha_{n-1}} + \dots + a_0 s^{\alpha_0}}. \quad (5)$$

单输入单输出分数阶控制系统的框图如图 1 所示。其中, r 为给定值, y 为被控对象输出值, e 为两者的偏差,分数阶 PID 控制器即根据偏差 e 调整控制器输出 u 。分数阶 PID 控制器的微分方程表达式为:

$$u(t) = K_p e(t) + K_i D^{-\alpha} e(t) + K_d D^{\beta} e(t), \quad (6)$$

其相应的控制器传递函数为:

$$G_c(s) = K_p + \frac{K_i}{s^\alpha} + K_d s^\beta. \quad (7)$$

其中, K_p 、 K_i 、 K_d 分别为比例、积分和微分增益, α 和 β 分别为积分和微分阶数。当 α 和 β 为 1 时,分数阶 PID 控制器退化为经典的整数 PID 控制器。与整数阶 PID 控制器相比,分数阶 PID 控制器中积分和微分的阶次可不为 1,从而为控制器设计增加了两个可调参数,使得分数阶 PID 控制器与整数阶 PID 控制器相比达到更好的控制性能指标。

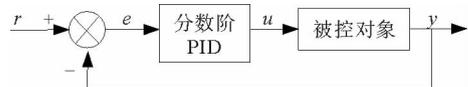


图 1 分数阶控制系统框图

Fig. 1 Block diagram of fractional-order control system

2 粒子群优化算法

粒子群优化算法是由 Kennedy 和 Eberhart 提出的一种模拟鸟群或鱼群群体行为的启发式智能优化算法^[10], 目前已经被广泛地应用于各类优化问题的求解。粒子群优化算法定义了一组与优化问题相关的粒子, 每一个粒子是待优化问题的一个可行解, 其维数与优化问题可行解的维数相同。粒子群优化算法根据各个粒子的历史最优解和粒子群中的全局最优解迭代更新每一个粒子的飞行速度, 并进而更新粒子位置, 最终得到优化问题的最优解。粒子群优化算法的速度更新方程和位置更新方程如下:

$$v_i(k+1) = \omega v_i(k) + c_1 r_1 (p_i(k) - x_i(k)) + c_2 r_2 (p_g(k) - x_i(k)); \quad (8)$$

$$x_i(k+1) = x_i(k) + v_i(k+1)。 \quad (9)$$

其中, ω 为惯性权系数, $v_i(k)$ 和 $x_i(k)$ 分别为粒子 i 在第 k 代的速度和位置, $p_i(k)$ 和 $p_g(k)$ 分别为粒子 i 的历史最优位置和当前代粒子群中的全局最优位置, 常数 c_1 和 c_2 称为加速系数, r_1 和 r_2 为 $[0, 1]$ 中均匀分布的随机数。

目前, 众多学者提出了一系列改进的 PSO 算法。如 Shi 等^[11] 提出了惯性权系数线性降低的 PSO-LDIW 算法, 其惯性权系数的计算方法为

$$\omega = (\omega_1 - \omega_2) \times \frac{k_{\max} - k}{k_{\max}} + \omega_2, \quad (10)$$

其中, k_{\max} 为最大迭代次数。Ratnaweera 等^[12] 提出了具有时变加速系数的 PSO-TVAC 算法, 其加速系数计算如下

$$c_1 = (c_{1i} - c_{1f}) \times \frac{k_{\max} - k}{k_{\max}} + c_{1f}, \quad (11)$$

$$c_2 = (c_{2i} - c_{2f}) \times \frac{k_{\max} - k}{k_{\max}} + c_{2f}。 \quad (12)$$

Clerk^[13] 为提高算法的搜索性能提出了具有压缩因子的 PSO-CK 算法等。此外, 除了对 PSO 算法中的参数选择提出改进方法之外, 结合多种辅助操作的混杂 PSO 算法也是学者们关注的研究方向^[14-16]。

3 分数阶 PID 控制器设计

本文使用 MATLAB/Simulink 仿真, 基于粒子群优化算法设计分数阶 PID 控制器。由于 MATLAB 目前尚没有正式推出面向分数阶系统仿真的工具箱, 因此分数阶 PID 控制器首先近似为整数阶系统, 然后可按照整数阶系统搭建仿真环境, 系统仿真框图如图 2 所示。

在仿真过程中, 本文使用 Oustaloup 近似化方法对分数阶 PID 控制器中的微积分算子进行近似。设待近似的微积分算子为 s^γ , 拟合频率段为 (ω_b, ω_h) , 则可以构造出一个近似 s^γ 的频域滤波器如下^[9]:

$$s^\gamma = K \prod_{k=1}^N \frac{s + \omega'_k}{s + \omega_k}, \quad (13)$$

$$K = \omega_h^\gamma, \quad (14)$$

$$\omega'_k = \omega_b \omega_u^{(2k-1-\gamma)/N}, \quad (15)$$

$$\omega_k = \omega_b \omega_u^{(2k-1+\gamma)/N}。 \quad (16)$$

本文的粒子群优化算法中的粒子为 5 维, 各维分别为 K_p 、 K_i 、 K_d 、 α 、 β 。优化过程中最大迭代次数为 100, 种群规模为 20, 按照式(10)~(12)计算惯性权系数和加速系数, 分数阶 PID 控制器设计的仿真流程如图 3 所示。

为验证本文的分数阶 PID 控制器设计方法的有效性, 以下使用文献[17]中的两个典型算例, 采用平方偏差积分(square deviation integral, ISE)、时间平方偏差积分(time squared deviation integral, ITSE)、绝对偏差积分(absolute deviation integral, IAE)和时间绝对偏差积分(time absolute deviation integral, ITAE)4 种性能指标设计分数阶 PID 控制器并进行比较。

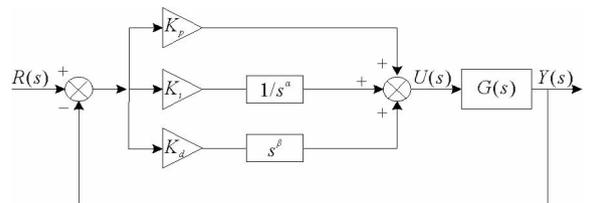


图 2 分数阶 PID 控制系统

Fig. 2 Fractional-order PID control system

例 1 高阶被控对象的传递函数为：

$$G(s) = \frac{1}{s^5 + 5s^4 + 10s^3 + 10s^2 + 5s + 1} \quad (17)$$

文献[17]中给出的控制器为：

$$C(s) = 0.7413(1 + 0.1869/s^{1.3106}) \quad (18)$$

根据文献[18]基于误差的最优控制器设计程序,对例 1 进行整数阶 PID 参数寻优,得到的控制器为：

$$C(s) = 1.534 + \frac{0.4561}{s} + 1.7347s \quad (19)$$

本文采用 4 种性能指标(ITSE、ITAE、ISE、IAE)得到的分数阶 PID 控制器参数如表 1 所示,图 4 给出了各控制器的阶跃响应,表 2 给出了相应控制器的性能指标。

由以上仿真结果可见,采用本文给出的参数优化方法得到的系统阶跃响应的超调量均不超过 10%,远远小于采用文献[17]和[18]方法得到的超调量;同时,除基于 ISE 指标的调节时间略大外,其余控制器性能指标均优于文献[17]和[18]中的设计方法;这说明本文给出的分数阶 PID 控制器设计方法比现有方法具有更好的性能。

表 1 例 1 分数阶 PID 控制器参数

Tab. 1 Parameters of fractional-order PID controllers of example 1

| | K_p | K_i | K_d | α | β |
|------|---------|---------|---------|----------|---------|
| ITSE | 1.663 7 | 0.337 8 | 2.000 0 | 1.054 5 | 1.120 6 |
| ITAE | 1.114 3 | 0.295 4 | 1.500 0 | 1.014 2 | 1.045 3 |
| ISE | 1.300 0 | 0.325 3 | 1.300 0 | 1.023 6 | 0.969 3 |
| IAE | 1.158 7 | 0.305 9 | 1.521 1 | 1.014 6 | 1.011 5 |

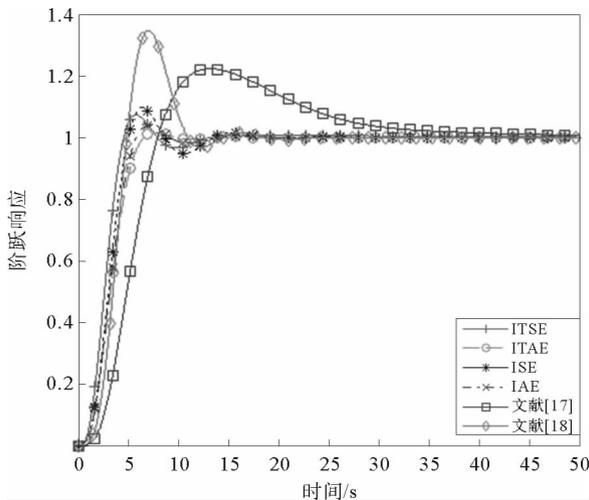


图 4 例 1 各控制器的阶跃响应

Fig. 4 Step response of controllers in Example 1

例 2 带时滞被控对象的传递函数为：

$$G(s) = \frac{1}{s^3 + 3s^2 + 3s + 1} e^{-s} \quad (20)$$

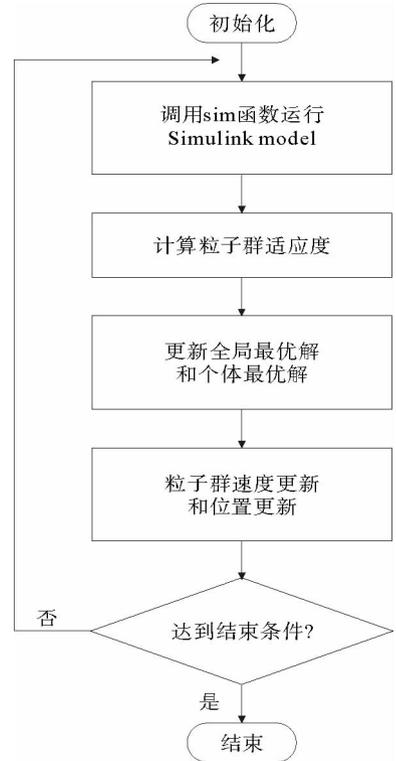


图 3 分数阶 PID 控制器设计仿真流程

Fig. 3 Simulation flowchart of fractional-order PID controller design

表 2 例 1 各控制器的性能指标

Tab. 2 Performance indexes of controllers in Example 1

| | 超调量 $M/\%$ | 调节时间 t_s/s | 峰值时间 t_p/s | 上升时间 t_r/s |
|--------|---------------|-----------------|-----------------|-----------------|
| ITSE | 7.4 | 6.789 | 5.812 | 4.593 |
| ITAE | 1.6 | 6.716 | 8.468 | 7.575 |
| ISE | 9.8 | 11.09 | 6.462 | 5.024 |
| IAE | 3.9 | 5.302 | 6.968 | 5.832 |
| 文献[17] | 22.4 | 27.48 | 13.5 | 7.958 |
| 文献[18] | 34.9 | 10.05 | 6.966 | 4.856 |

文献[17]中给出的控制器为:

$$C(s) = 0.7870 \left(1 + \frac{0.2852}{s^{1.3789}} \right) \quad (21)$$

根据文献[18]中基于误差的最优控制器设计程序,对例 2 进行整数阶 PID 参数寻优,得到的控制器为:

$$C(s) = 1.523 + \frac{0.4084}{s} + 0.9728s \quad (22)$$

本文采用四种性能指标(ITSE、ITAE、ISE、IAE)得到的分数阶 PID 控制器参数如表 3 所示,图 5 给出了各控制器的阶跃响应,表 4 给出了相应控制器的性能指标。由以上仿真结果可见,对含有时滞的被控对象,采用本文给出的参数优化方法能够克服文献[17]和[18]中方法的缺陷,所得到的系统阶跃响应具有更优越的性能。

表 3 例 2 分数阶 PID 控制器参数

Tab. 3 Parameters of fractional-order PID controllers of Example 2

| | K_p | K_i | K_d | α | β |
|------|--------|--------|--------|----------|---------|
| ITSE | 1.5 | 0.4351 | 1.5 | 1.0221 | 1.1584 |
| ITAE | 1.3315 | 0.4389 | 1.4542 | 1.0123 | 1.0966 |
| ISE | 1.5 | 0.4494 | 1.5 | 0.9952 | 1.2065 |
| IAE | 1.5 | 0.4369 | 1.5 | 1.0181 | 1.1532 |

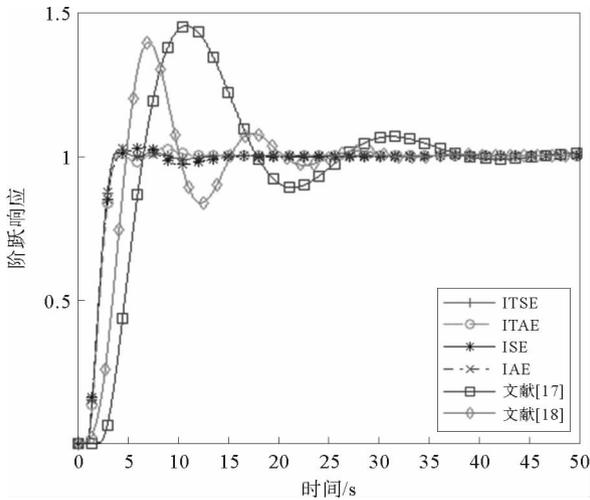


图 5 例 2 各控制器的阶跃响应

Fig. 5 Step response of controllers in Example 2

表 4 例 2 各控制器的性能指标

Tab. 4 Performance indexes of controllers in Example 2

| | 超调量 $M/\%$ | 调节时间 t_s/s | 峰值时间 t_p/s | 上升时间 t_r/s |
|--------|---------------|-----------------|-----------------|-----------------|
| ITSE | 2.1 | 3.321 | 4.165 | 3.702 |
| ITAE | 2.4 | 3.465 | 8.511 | 3.938 |
| ISE | 3 | 3.509 | 6.429 | 4.247 |
| IAE | 2.6 | 3.302 | 4.291 | 3.646 |
| 文献[17] | 45.2 | 33.97 | 10.82 | 6.563 |
| 文献[18] | 29 | 12.85 | 5.192 | 3.873 |

从仿真结果来看,本文给出的基于粒子群优化算法的分数阶 PID 控制器设计方法在 4 种性能指标下均能够得到较好的控制器参数,阶跃响应具有响应速度快、超调量小、稳定时间短的特点,与文献[17]及文献[18]中的设计方法相比能够实现更好的控制性能。

4 结论

给出了一种基于粒子群优化算法的分数阶 PID 控制器设计方法,获得的控制器参数能够实现更好的控制性能,并对两个算例通过 MATLAB/Simulink 仿真验证了设计方法的有效性。本文可作为基于智能优化方法的分数阶 PID 控制器设计的一般框架,通过修改所使用的智能优化方法,可验证新算法性能并进行比较。在此基础上,将进一步研究自适应的分数阶 PID 控制器设计方法及数字分数阶 PID 控制器设计及其应用等。

参考文献:

- [1] CHEN Y Q, PETRAS I, XUE D. Fractional order control; A tutorial[C]//American Control Conference, Missouri; IEEE Press, 2009; 1397-1441.
- [2] 陈超波, 胡莉, 马颖, 等. 一种改进式遗传算法优化分数阶PID控制器[J]. 西安工业大学学报, 2016, 36(6): 510-516.
CHEN Chaobo, HU Li, MA Ying, et al. Improved genetic algorithm for optimal fractional-order PID controller[J]. Journal of Xi'an Institute of Technology, 2016, 36(6): 510-516.
- [3] ZHANG D L, TANG Y G, GUAN X P. Optimum design of fractional order PID controller for an AVR system using an improved artificial bee colony algorithm[J]. Acta Automatica Sinica, 2014, 40(5): 973-980.
- [4] ATES A, YEROGLU C. Optimal fractional order PID design via Tabu Search based algorithm[J]. ISA Transactions, 2016, 60: 109-118.
- [5] MOHARAM A, EL-HOSSEINI M A, ALI H A. Design of optimal PID controller using hybrid differential evolution and particle swarm optimization with an aging leading and challengers[M]. North-holland; Elsevier Science Publisher B. V, 2016, 38: 727-737.
- [6] LIU X. Optimization design on fractional order PID controller based on adaptive particle swarm optimization algorithm[J]. Nonlinear Dynamics, 2016, 84(1): 379-386.
- [7] CHEN Y Q, VINAGRE B M. A new IIR-type digital fractional order differentiator[J]. Signal Processing, 2003, 83(11): 2359-2365.
- [8] VINAGRE B M, PODLUBNY I, HERNANDEZ A, et al. Some approximations of fractional order operators used in control theory and applications[J]. Fractional Calculus & Applied Analysis, 2000, 3(3): 231-248.
- [9] MONJE C A, CHEN Y Q, VINAGRE B M, et al. Fractional-order systems and controls[M]. London: Springer Press, 2010.
- [10] KENNEDY J, EBERHART R. Particle swarm optimization[C]//IEEE International Conference on Neural Networks, Perth; IEEE Press, 2002, 4: 1942-1948.
- [11] SHI Y, EBERHART R. Modified particle swarm optimizer[C]//IEEE International Conference on Evolutionary Computation Proceeding, Alaska; IEEE Press, 1998: 69-73.
- [12] RATNAWEERA A, HALGAMUGE S K, WATSON H C. Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients[J]. IEEE Transactions on Evolutionary Computation, 2004, 8: 240-255.
- [13] CLERC M, KENNEDY J. The particle swarm: Explosion, stability, and convergence in a multi-dimensional complex space [J]. IEEE Transactions on Evolutionary Computation, 2002, 6(1): 58-73.
- [14] ANDREWS P S. An investigation into mutation operators for particle swarm optimization[C]//Proceedings of IEEE Congress on Evolutionary Computation, Vancouver; IEEE Press, 2006: 1044-1045.
- [15] MENDES R, KENNEDY J, NEVES J. The fully informed particle swarm; Simpler, maybe better[J]. IEEE Transactions on Evolutionary Computation, 2004, 8(3): 204-210.
- [16] ZHANG W J, XIE X F. DEPSO: Hybrid particle swarm with differential evolution operator[C]// Proceedings of IEEE International Conference on Systems, Man, and Cybernetics, Washington; IEEE Press, 2004: 997-1006.
- [17] 王春阳, 李明秋, 姜淑华, 等. 分数阶控制系统设计[M]. 北京: 国防工业出版社, 2014.
- [18] 薛定宇, 陈阳泉. 控制数学问题的 MATLAB 求解[M]. 北京: 清华大学出版社, 2007.

(责任编辑: 高丽华)