ICIC Express Letters ICIC International ©2016 ISSN 1881-803X
Volume **10**, Number **9**, September **2016** pp. 2191–2196

DESIGN OF FRACTIONAL-ORDER PID CONTROLLERS FOR FRACTIONAL-ORDER SYSTEMS: A BINARY-CODED INDIVIDUAL-BASED EXTREMAL OPTIMIZATION METHOD

Di Wu, Huan Wang, Guoqiang Zeng* and Kangdi Lu

College of Physics & Electronic Information Engineering
Wenzhou University
Higher Educational Park, Chashan, Wenzhou 325035, P. R. China
*Corresponding author: zeng.guoqiang5@gmail.com; zenggq@wzu.edu.cn

Received January 2016; accepted April 2016

Abstract. *The issue of how to design an effective and efficient fractional-order PID (FOPID) controller for a fractional-order industrial control system to obtain optimal performances is still an open topic. From the perspective of evolutionary algorithms, this paper presents a binary-coded individual-based extremal optimization (BIEO) method to deal with this issue. The proposed method encodes five FOPID parameters into a binary string, evaluates the control performance in terms of a more reasonable index than the integral of absolute error (IAE) by considering the trade-off between steady-state and transient-state performance and updates the solution by using the individual-based evolutionary mechanism consisting of the power-law probability distribution based selection and binary mutation for the selected bad elements. The proposed BIEO method has only selection and mutation operations, and it is simpler than binary-coded adaptive genetic algorithm (GA), and particle swarm optimization (PSO) due to its fewer adjustable parameters. Furthermore, the superiority of BIEO-FOPID to other popular evolutionary algorithms-based FOPID methods and analytic methods is demonstrated by the experimental results on some typical fractional-order control systems.*
Keywords: Fractional-order PID controller, Fractional-order control system, Extremal optimization, Evolutionary algorithms

1. **Introduction.** As a generalization of a standard PID controller based on fractional-order calculus, fractional-order PID (FOPID) controller namely PI$^\lambda$D$^\mu$ controller was firstly proposed by Podlubny [1], which has been demonstrated to provide better control performance than standard PID controller due to extra degrees of freedom introduced by an integrator of fractional order λ and a differentiator of fractional order μ. Consequently, FOPID controllers have attracted increasing attention by the academic and industrial communities [2-6]. How to design and tune an optimal FOPID controller to obtain high-quality performances, such as high stability, satisfied transient response, excellent steady performance and good robustness, is of great theoretical and practical significance, but is still an open issue. In the attempt to address this issue, some researchers have made a great deal of effort from different perspectives of analytic methods [3,4] and evolutionary algorithms-based methods [5,6]. However, there are only few research works concerning design methods of FOPID controllers for fractional-order industrial control systems.

Extremal optimization (EO) [7] is a novel meta-heuristics optimization framework originally inspired by self-organized criticality (SOC) [8]. In the past decade, the basic EO algorithm and its modified versions have been successfully applied to a variety of benchmark and real-world engineering optimization problems [9-12]. The more comprehensive introduction concerning EO is referred to in the surveys [13]. However, the applications of EO to the design of PID are relatively rare [14,15]. To the best of our knowledge, there is no reported research work concerning the optimum design of FOPID controllers for fractional-order industrial control systems from the perspective of EO.

In this letter, we propose a binary-coded individual-based extremal optimization (BIEO) method to design FOPID controllers for fractional-order industrial control systems. The rest of the paper is organized as follows. In Section 2, we will present preliminaries on fractional-order control systems. Section 3 provides the details of the BIEO-FOPID method. In Section 4, we give the comparative results between the proposed method and other evolutionary algorithms for typical fractional-order control systems. Finally, Section 5 concludes this paper and suggests direction for future work.

2. Preliminaries.

Definition 2.1. *The transfer function $G_{fc}(s)$ of an FOPID controller is defined as follows [1]:*

$$G_{fc}(s) = \frac{U_f(s)}{E_f(s)} = K_P + K_I s^{-\lambda} + K_D s^{\mu} \tag{1}$$

where K_P, K_I, and K_D are proportional, integral, derivative gain, respectively, and λ, μ are the fractional-order parameters of integrator and differentiator, respectively, and $\lambda > 0$, $\mu > 0$. Note that the standard integer order PID controller is one of the special FOPID controllers with $\lambda = 1$ and $\mu = 1$.

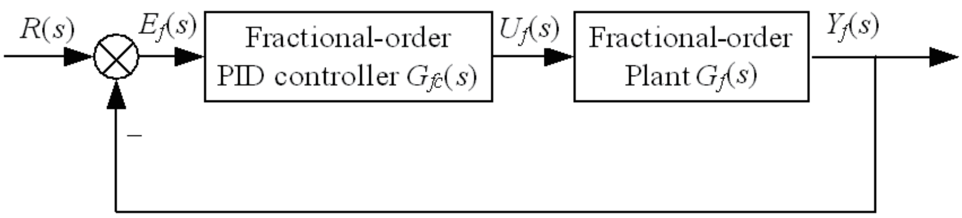

Figure 1. Block diagram of a fractional-order control system with an FOPID controller

3. The Proposed Method.

In this letter, a novel performance criterion is proposed to evaluate an FOPID controller by considering not only IAE, but also the following factors. More accurately, the definition of the proposed performance criterion is presented as follows.

Definition 3.1. *For a binary-coded solution $S = [K_P, K_I, K_D, \lambda, \mu]$, which represents an FOPID controller, the corresponding performance criterion $F(S)$ in the time domain is defined as follows:*

$$F(S) = \begin{cases} w_1 M_p + w_2(t_r + t_s) + w_3 E_{ss} + \int_0^{\infty} (w_4 \left|e_f(t)\right|)\, dt, & \text{if} \quad \Delta y(t) \geq 0 \\ w_1 M_p + w_2(t_r + t_s) + w_3 E_{ss} + \int_0^{\infty} (w_4 \left|e_f(t)\right| + w_5 \left|\Delta y(t)\right|)\, dt, & \text{if} \quad \Delta y(t) < 0 \end{cases} \tag{2}$$

where M_p, E_{ss}, t_r, t_s are overshoot, steady-state error, rise time, and settling time, respectively, $e(t)$ is the system error, $\Delta y(t) = y(t) - y(t - \Delta t)$, $u(t)$ is the control output at the time t, and w_1, w_2, w_3, w_4, w_5 are weight coefficients.

The BCEO-based FOPID controller design method for a fractional-order control system is presented as Algorithm 3.1.

Algorithm 3.1

Input: A fractional-order plant $G_f(s)$ with an FOPID controller and sampling period T_s, the length l_j of j-th binary substring corresponding to j-th parameter, the weight coefficients w_1, w_2, w_3, w_4, w_5 used for evaluating the fitness, the maximum number of iterations $I_{\max}$, and the adjustable parameter τ of power-law distribution $P(k)$.

Output: The best solution S_{best} (the best FOPID parameters K_{PO}, K_{IO}, K_{DO}, λ_O, μ_O) and the corresponding global fitness F_{best}.

1. Generate an initial solution S randomly, where $S = [K_P, K_I, K_D, \lambda, \mu]$ is a binary string with length $L = \sum_{j=1}^{5} l_j$ that encodes five FOPID design parameters and set $S_{\text{best}} = S$ and $F_{\text{best}} = F(S)$ according to Definition 3.1;
2. Generate the configuration S_i by flipping the bit i ($1 \leq i \leq L$) and keeping the others unchanged for solution S, and then compute the fitness $F(S_i)$ by Definition 3.1;
3. Evaluate the local fitness $\lambda_i = F(S_i) - F_{\text{best}}$ for each bit i and rank all the bits according to λ_i, i.e., find a permutation Π_1 of the labels i such that $\lambda_{\Pi_1(1)} \geq \lambda_{\Pi_1(2)} \geq \cdots \geq \lambda_{\Pi_1(L)}$;
4. Select a rank $\Pi_1(k)$ according to a probability distribution $P(k) \propto k^{-\tau}$, ($1 \leq k \leq L$), and denote the corresponding bit as x_j;
5. Flip the value of x_j and set $S_{\text{new}} = S$ in which x_j value is flipped;
6. If $F(S_{\text{new}}) \leq F(S_{\text{best}})$, then $S_{\text{best}} = S_{\text{new}}$;
7. Accept S_{new} unconditionally;
8. Repeat step (2) to step (7) until some predefined stopping criteria (e.g., the maximum number of iterations) are satisfied;
9. Obtain the best solution S_{best}, the corresponding global fitness F_{best}, and the corresponding system output y_b.

From the above description, it is clear that the proposed BIEO-FOPID method has only selection and mutation operations, and it is simpler than GA and PSO due to its fewer adjustable parameters.

4. **Experimental Results and Discussion.** In order to demonstrate the superiority of the proposed BIEO-FOPID method, this section provides the comparative results between BIEO-FOPID and other evolutionary algorithms-based FOPID and PID, e.g., binary-coded adaptive GA (BAGA) [16] and real-coded PSO (RPSO) [5], for some typical fractional-order control systems. The fractional-order plant models of heating furnace and a typical industrial system are described as follows [3]:

$$G_{f1} = \frac{1}{14994s^{1.31} + 6009.5s^{0.97} + 1.69} \tag{3}$$

$$G_{f2} = \frac{1}{0.8s^{2.2} + 0.5s^{0.9} + 1} \tag{4}$$

Equations (3) and (4) are denoted as FOP1 and FOP2, respectively. In the experiments, the weight coefficients are set as follows: $w_1 = 1$, $w_2 = 2$, $w_3 = 1000$, $w_4 = 1$ and $w_5 = 100$. l_j are set as 10 for all five FOPID parameters. τ are set as 1.25 and 1.30, and T_s are set as 1.5 and 0.005 seconds for FOP1 and FOP2, respectively. Table 1 presents the comparative results of the best (f_b), average (f_a), worst (f_w) fitness, standard deviation (SD) and average computational time (T_a) obtained by independent 20 runs for each algorithm. It is clear that f_a and f_b of BIEO-FOPID are all better than those of BAGA-FOPID/PID, RPSO-FOPID/PID, and BIEO-PID. Furthermore, the average computational time of BIEO-FOPID is also smaller than that of BAGA-FOPID/PID and RPSO-FOPID/PID.

The performance corresponding to the best fitness obtained by BIEO-FOPID and other evolutionary algorithms-based FOPID and PID for FOP1 and FOP2 is shown in Table 2. Clearly, BIEO-FOPID performs better than Analysis-FOPID [3], BAGA-FOPID/PID [16], RPSO-FOPID/PID [5] and BIEO-PID in terms of $M_p(\%)$, t_r and $t_s(0.5\%)$ for FOP1,

TABLE 1. The comparative fitness, SD and average computational time between BIEO-FOPID and other evolutionary algorithms-based FOPID and PID

Plant	Algorithm	f_w	f_a	f_b	SD	T_a
FOP1	BIEO-FOPID	32.80114	**27.95568**	**26.99791**	1.255641	**61.34**
	BAGA-FOPID [16]	35.25564	30.26548	27.15659	2.311658	67.52
	RPSO-FOPID [5]	31.51506	28.24235	27.44610	0.982803	66.27
	BIEO-PID	38.52643	36.70688	35.89686	1.040050	61.51
	BAGA-PID [16]	49.43986	38.36536	35.89256	3.185046	67.60
	RPSO-PID [5]	38.04790	36.61270	35.94730	0.525652	66.32
FOP2	BIEO-FOPID	1.693123	**1.432061**	**1.315507**	0.105897	**62.65**
	BAGA-FOPID [16]	2.231206	1.492074	1.326255	0.188245	68.17
	RPSO-FOPID [5]	1.511396	1.460190	1.333455	0.080170	66.89
	BIEO-PID	9.784083	9.781088	9.780728	0.000885	62.70
	BAGA-PID [16]	10.43481	9.925076	9.789330	0.126009	68.35
	RPSO-PID [5]	10.25623	9.968738	9.798902	0.096361	66.91

TABLE 2. The performance corresponding to the best fitness obtained by BIEO-FOPID and other evolutionary algorithms-based FOPID and PID for FOP1 and FOP2

Plant	Algorithm	f_b	$M_p(\%)$	t_r	$t_s(0.5\%)$	E_{ss}
FOP1	BIEO-FOPID	**26.99791**	**0.566**	**6**	**15**	1.47E-04
	BAGA-FOPID [16]	27.15659	0.654	6	15	1.26E-04
	RPSO-FOPID [5]	27.44610	0.795	6	16.5	1.37E-04
	Analysis-FOPID [3]	327.0014	8.686	43.5	177	9.18E-04
	BIEO-PID	35.89686	1.870	7.5	13.5	6.50E-05
	BAGA-PID [16]	35.89256	1.867	7.5	13.5	6.20E-05
	RPSO-PID [5]	35.94730	1.887	7.5	13.5	1.61E-05
FOP2	BIEO-FOPID	**1.315507**	**0.00015**	**0.01**	**0.055**	**2.12E-07**
	BAGA-FOPID [16]	1.326255	0.00044	0.01	0.055	4.36E-06
	RPSO-FOPID [5]	1.333455	0.00667	0.01	0.055	3.28E-06
	Analysis-FOPID [3]	111.4477	31.812	0.115	1.50	7.41E-03
	BIEO-PID	9.780728	3.472	0.02	0.10	5.70E-04
	BAGA-PID [16]	9.789330	3.478	0.02	0.10	5.68E-04
	RPSO-PID [5]	9.798902	3.503	0.02	0.10	5.42E-04

and it performs better than above methods in terms of $M_p(\%)$, t_r, $t_s(0.5\%)$ and E_{ss} for FOP2.

To illustrate the good convergence characteristic of the proposed BIEO-FOPID, Figure 2 presents dynamical optimization process of the best fitness obtained by BIEO-FOPID and other evolutionary algorithms-based FOPID/PID for FOP1 and FOP2. The system outputs corresponding to best fitness obtained by BIEO-FOPID and other evolutionary algorithms-based FOPID/PID and Analysis-FOPID for FOP1 and FOP2 are shown as Figure 3.

5. **Conclusion.**

This letter presents a binary-coded individual-based extremal optimization (BIEO) method to design FOPID controllers for fractional-order industrial control systems. The proposed BIEO-FOPID is not only simpler than BAGA-FOPID and RPSO-FOPID, but also is better than BAGA-FOPID/PID, RPSO-FOPID/PID and BIEO-PID in terms of transient and steady-state performance indices. In fact, the performance of

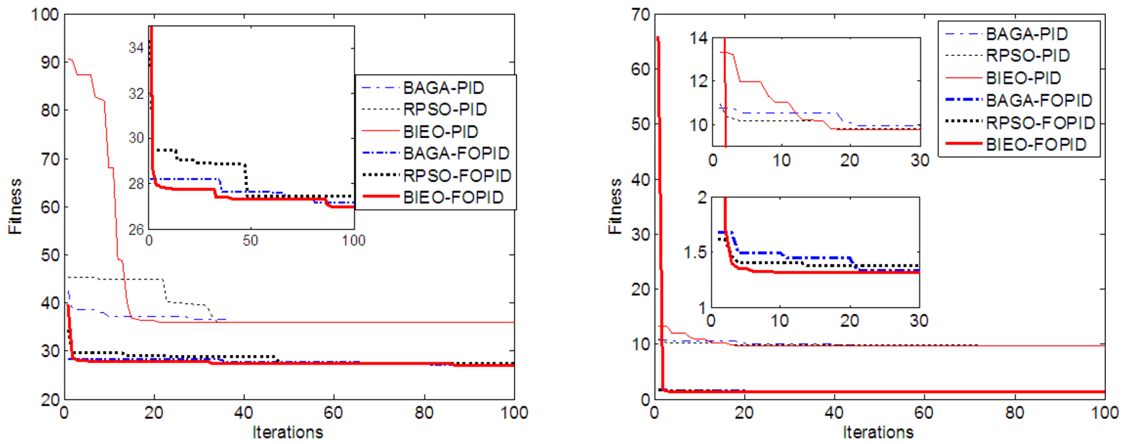

FIGURE 2. The dynamical optimization process of the best fitness obtained by BIEO-FOPID and other evolutionary algorithms-based FOPID and PID for FOP1 (left) and FOP2 (right)

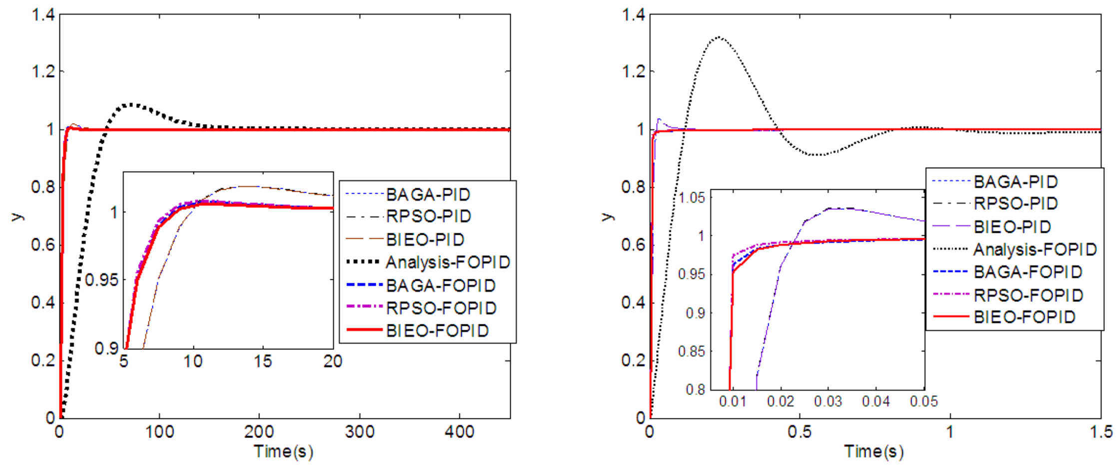

FIGURE 3. Comparison of system outputs corresponding to best fitness obtained by BIEO-FOPID and other algorithms-based FOPID and PID for FOP1 (left) and FOP2 (right)

BIEO-FOPID can be further improved by using adaptive probability distribution based evolutionary mechanism. Furthermore, the basic idea behind BIEO-FOPID will be extended to more complex engineering systems.

Acknowledgment. The authors gratefully acknowledge the helpful comments and suggestions of the editor and anonymous reviewers. This work is supported partially by the National Natural Science Foundation of China (No. 51207112), Science and Technology Planning Project of Zhejiang Province (Nos. 2014C31074, 2014C31093, 2015C31157), and Zhejiang Provincial Natural Science Foundation of China (Nos. LY16F030011, LZ16E050002, LQ14F030006, LQ14F030007).

REFERENCES

[1] I. Podlubny, Fractional-order systems and PI$^{\lambda}$D$^{\mu}$ controllers, *IEEE Trans. Automatic Control*, vol.44, no.1, pp.208-213, 1999.
[2] C. A. Monje, Y. Q. Chen, B. M. Vinagre, D. Y. Xue and V. Feliu, *Fractional-Order Systems and Controls: Fundamentals and Applications*, Springer, 2010.

[3] D. Y. Xue and C. N. Zhao, Fractional order PID controller design for fractional order system, *Control Theory & Applications*, vol.24, no.5, pp.771-776, 2007.
[4] Z. Gao, M. Yan and J. X. Wei, Robust stabilizing regions of fractional-order PD^{μ} controllers of time-delay fractional-order systems, *Journal of Process Control*, vol.24, pp.37-47, 2014.
[5] M. Zamani, M. K. Ghartemani, N. Sadati and M. Parniani, Design of a fractional order PID controller for an AVR using particle swarm optimization, *Control Engineering Practice*, vol.17, pp.1380-1387, 2009.
[6] Y. G. Tang, M. Y. Cui, C. C. Hua, L. X. Li and Y. X. Yang, Optimum design of fractional order $\mathrm{PI}^{\lambda}\mathrm{D}^{\mu}$ controller for AVR system using chaotic ant swarm, *Expert Systems with Applications*, vol.39, pp.6887-6896, 2012.
[7] S. Boettcher and A. G. Percus, Optimization with extremal dynamics, *Physical Review Letters*, vol.86, no.23, pp.5211-5214, 2001.
[8] P. Bak, C. Tang and K. Wiesenfeld, Self-organized criticality, *Physical Review Letters*, vol.59, pp.381-384, 1987.
[9] G. Q. Zeng, J. Chen, L. M. Li, M. R. Chen, L. Wu, Y. X. Dai and C. W. Zheng, An improved multi-objective population-based extremal optimization algorithm with polynomial mutation, *Information Science*, vol.330, pp.49-73, 2016.
[10] L. M. Li, K. D. Lu, G. Q. Zeng, L. Wu and M. R. Chen, A novel real-coded population-based extremal optimization algorithm with polynomial mutation: A non-parametric statistical study on continuous optimization problems, *Neurocomputing*, vol.174, pp.577-587, 2016.
[11] G. Q. Zeng, Y. Z. Lu, Y. X. Dai, Z. G. Wu, W. J. Mao, Z. J. Zhang and C. W. Zheng, Backbone guided extremal optimization for the hard maximum satisfiability problem, *International Journal of Innovative Computing, Information and Control*, vol.8, no.12, pp.8355-8366, 2012.
[12] G. Q. Zeng, Z. J. Zhang, Y. X. Dai and C. W. Zheng, Modified extremal optimization with efficient initial configurations and extended probability distributions-based evolutionary mechanism, *ICIC Express Letters, Part B: Applications*, vol.3, no.6, pp.1631-1636, 2012.
[13] G. Q. Zeng and Y. Z. Lu, Survey on computational complexity with phase transitions and extremal optimization, *Proc. of the 48th IEEE conf. Control and Decision & the 28th Chinese Control Conf.*, pp.4352-4359, 2009.
[14] G. Q. Zeng, K. D. Lu, Y. X. Dai, Z. J. Zhang, M. R. Chen, C. W. Zheng, D. Wu and W. W. Peng, Binary-coded extremal optimization for the design of PID controllers, *Neurocomputing*, vol.138, pp.180-188, 2014.
[15] G. Q. Zeng, J. Chen, M. R. Chen, Y. X. Dai, L. M. Li, K. D. Lu and C. W. Zheng, Design of multi-variable PID controllers using real-coded population-based extremal optimization, *Neurocomputing*, vol.151, pp.1343-1353, 2015.
[16] J. Zhang, S. H. Chung and W. L. Lo, Clustering-based adaptive crossover and mutation probabilities for genetic algorithms, *IEEE Trans. Evolutionary Computation*, vol.11, pp.326-335, 2007.