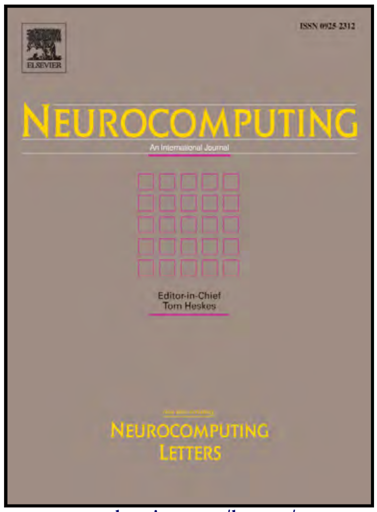


Design of fractional order PID controller for automatic regulator voltage system based on multi-objective extremal optimization

Guo-Qiang Zeng, Jie Chen, Yu-Xing Dai, Li-Min Li, Chong-Wei Zheng, Min-Rong Chen

PII: S0925-2312(15)00221-0
DOI: http://dx.doi.org/10.1016/j.neucom.2015.02.051
Reference: NEUCOM15188

To appear in: *Neurocomputing*

Received date: 17 December 2014
Revised date: 18 January 2015
Accepted date: 13 February 2015

Design of fractional order PID controller for automatic regulator voltage system based on multi-objective extremal optimization

Guo-Qiang Zeng[1,*], Jie Chen[1], Yu-Xing Dai[1], Li-Min Li[1], Chong-Wei Zheng[1], Min-Rong Chen[2]

[1]*Department of Electrical and Electronic Engineering, Wenzhou University, Wenzhou, 325035, China*

[2]*College of Information Engineering, Shenzhen University, Shenzhen, 518060, China*

*Corresponding author: Guo-Qiang Zeng, *E-mail*:{zeng.guoqiang5@gmail.com}

Abstract: Design of an effective and efficient fractional order PID (FOPID) controller, as a generalization of a standard PID controller based on fractional order calculus, for an industrial control system to obtain high-quality performances is of great theoretical and practical significance. From the perspective of multi-objective optimization, this paper presents a novel FOPID controller design method based on an improved multi-objective extremal optimization (MOEO) algorithm for an automatic regulator voltage (AVR) system. The problem of designing FOPID controller for AVR is firstly formulated as a multi-objective optimization problem with three objective functions including minimization of integral of absolute error (IAE), absolute steady-state error, and settling time. Then, an improved MOEO algorithm is proposed to solve this problem by adopting individual-based iterated optimization mechanism and polynomial mutation (PLM). From the perspective of algorithm design, the proposed MOEO algorithm is relatively simpler than NSGA-II and single-objective evolutionary algorithms, such as genetic algorithm (GA), particle swarm optimization (PSO), chaotic anti swarm (CAS) due to its fewer adjustable parameters. Furthermore, the superiority of proposed MOEO-FOPID controller to NSGA-II-based FOPID, single-objective evolutionary algorithms-based FOPID controllers, MOEO-based and NSGA-II-based PID controllers is demonstrated by extensive experimental results on an AVR system in terms of accuracy and robustness.

Keywords: Extremal optimization; Multi-objective optimization; Fractional order PID controller; Automatic regulator voltage system

1. Introduction

In the past decades, a great many advancements have been gained in control theories and practices [1]-[4], proportional-integral-derivative (PID) control is still widely recognized as one of the simplest yet most effective control strategies in the control industry [5]-[10]. 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 [11], and it has been demonstrated to provide better control performance than standard integer order PID controller due to extra degrees of freedom introduced by an integrator of fractional order λ and a differentiator of fractional order μ. As a consequence, FOPID controller has attracted increasing attentions by the academic and industrial community [12]-[17]. On the other hands, the introduction of extra parameters in a FOPID controller also increases the difficulty of tuning satisfied values of parameters, so 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 far from well-understood. In the attempt to address this issue, some researchers have made a great deal of efforts from the following different respective of analytical methods [18]-[22] and evolutionary algorithms-based methods [14], [23]-[28]. More specifically, the evolutionary algorithms, such as genetic algorithm (GA) [14], chaotic ant swarm (CAS) [14], particle swarm optimization (PSO) [23], [24], differential evolution (DE) [25], artificial been colony algorithm [26], hybrid algorithm combing with electromagnetism-like algorithm and GA [27], have been utilized for the design of FOPID controllers. Nevertheless, most of the reported research works focus on single-objective optimization for the design of FOPID controllers. In practice, multi-objective optimization algorithms [28]-[31] are required to design FOPID and PID controllers because of contradictory objective functions and performance metrics, e.g., integral of the time multiplied squared error (ITSE) and the integral of the squared deviation of controller output (ISDCO) [30]. However, the reported studies concerning design of FOPID controller based on multi-objective evolutionary algorithms (MOEAs) is considered as just a beginning because only NSGA-II [32] has been extended to design FOPID controllers so far. This paper presents an alternative effective MOEA method based on multi-objective extremal optimization called MOEO for the design of FOPID controller in an automatic regulator voltage (AVR) [33] system, which is used to maintain the terminal voltage of a synchronous generator at a desired level.

As a novel evolutionary algorithm originally inspired by far-from-equilibrium dynamics of self-organized criticality (SOC) [34],[35], extremal optimization (EO) [36]-[38] provides a novel insight into optimization domain because it merely selects against the bad instead of favoring the good randomly or according to a power-law probability distribution [39]. The mechanism of EO can be characterized from the perspectives of statistical physics, biological co-evolution and ecosystem [40]. So far, the EO algorithm and its modified versions have been successfully applied to a variety of benchmark and real-world engineering optimization problems, such as graph partitioning [41], graph coloring [42], travelling salesman problem [43], [44], maximum satisfiability (MAX-SAT) problem [45], [46], numerical optimization problems and multi-objective optimization problems [47], community detection in complex network [48], steel production scheduling [49], design of heat pipe [50], and unit commitment problem for power systems [51]. The more comprehensive introduction concerning EO is referred to the surveys [52], [53]. Unfortunately, there are only few reported research works concerning the applications of EO in the field of multi-objective optimization [54]-[56]. Chen and Lu [55] propose an individual elitist ($1+\lambda$) multi-objective algorithm called multi-objective extremal optimization (MOEO) based on a single solution, in which a new hybrid mutation operator combining Gaussian mutation with Cauchy mutation to enhances the exploratory capabilities. In [56], another Pareto-based algorithm named Multi-objective Population-based Extremal Optimization (MOPEO), which adopts population-based iterated mechanism and non-uniform mutation. Although these works only focus on some benchmark multi-objective optimization functions, e.g., ZDT1, ZDT2, ZDT3, ZDT4, and ZDT6, the experimental results on these benchmark problems have shown that MOPEO [55] and MOEO [56] provide better performance than the popular NSGA-II [32] and some other reported MOEAs. This is one of the primary motivations to extend MOEO algorithms to design FOPID controllers for an AVR system.

To the best of our knowledge, this paper is the first reported research work concerning multi-objective EO for the optimal design FOPID and PID controllers. The key idea behind the proposed method is formulating the FOPID design problem for AVR system as a multi-objective optimization problem with three objective functions including integral of absolute error (IAE), absolute steady-state error, and settling time, and solving this problem by developing an improved MOEO algorithm, which consists of the following main components, such as generation of a random real-coded individual representing a FOPID controller, updating the current individual based on polynomial mutation [60], Pareto-based fitness assignment strategy based on non-dominated sorting, updating the external archive according to the archive controller and the crowding-distance metric. In comparison to NSGA-II-based FOPID design algorithm [30], the proposed MOEO algorithm adopts individual-based iterated optimization mechanism with only mutation operation called polynomial mutation. From the perspective of algorithm design, the proposed MOEO algorithm is relatively simpler than NSGA-II [30] and reported competitive single-objective evolutionary algorithms, such as GA [14], PSO [14],[24], CAS [14] due to its fewer adjustable parameters and single individual-based iterated optimization mechanism with only mutation operation. Furthermore, extensive experimental results on AVR system have shown that the proposed MOEO-FOPID controller is superior to NSGA-II-FOPID, single-objective evolutionary algorithms-based FOPID controllers, MOEO-based and NSGA-II-based PID controllers in terms of accuracy and robustness.

The rest of this paper is organized as follows. Section 2 presents preliminaries concerning fractional order PID controller, AVR system, multi-objective optimization problems and EO. The MOEO-based FOPID design algorithm is proposed in section 3. The experimental results on AVR system are given and discussed in section 4. Finally, we give the conclusion and open problems in section 5.

2. Preliminaries

2.1. Fractional order PID controller

As one the most commonly used definitions for fractional differ-integral, Riemann-Liouville (RL) definition are given as the following form [57]:

$$ {}_aD_t^r f(t) = \frac{1}{\Gamma(n-r)} \frac{d^n}{dt^n} \int_a^t \frac{f(\tau)}{(t-\tau)^{r-n+1}} d\tau,\ n-1 < r < n \tag{1} $$

where $\Gamma(.)$ is the Gamma function. The Laplace transform of RL fractional derivative (1) is expressed as follows:

$$ \int_0^\infty e^{-st}\, {}_0D_t^r f(t) dt = s^r F(s) - \sum_{k=0}^{n-1} s^k\, {}_0D_t^{r-k-1} f(t)dt \Big|_{t=0} \tag{2} $$

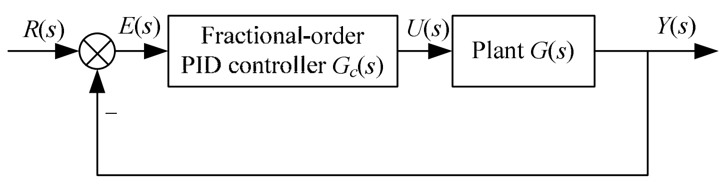


Fig.1. Block diagram of a control system with a FOPID controller

Fig.1 shows block diagram of a control system with a FOPID controller, which is also called $PI^{\lambda}D^{\mu}$ controller. Its definition in terms of transfer function is given as follows:

Definition 1 [11]: The transfer function $G_c(s)$ of a FOPID controller is defined in the equation (3):

$$ G_c(s) = \frac{U(s)}{E(s)} = K_P + K_I s^{-\lambda} + K_D s^{\mu} \tag{3} $$

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

From the perspective of time domains, the $PI^{\lambda}D^{\mu}$ controller is also expressed as the following form:

$$ u(t) = K_p e(t) + K_i D^{-\lambda} e(t) + K_d D^{\mu} e(t) \tag{4} $$

2.2. AVR system

An AVR system [33] consists of four main components including amplifier, exciter, generator, and sensor, and more details concerning the transfer functions with the range of parameters modeling these components are shown in Table 1. Here, K_A, K_E, K_G, and K_R are the gains of amplifier, exciter, generator, and sensors, respectively, and τ_A, τ_E, τ_G, and τ_R are inertia time constants of amplifier, exciter, generator, and sensors, respectively. The block diagram of an AVR system with a FOPID controller is given in Fig.2, where $V_{ref}(s)$ and $V_t(s)$ are the reference voltage and terminal voltage, respectively.

Table 1: Models of the components in an AVR system

Component	Transfer function	Parameters range
Amplifier	$K_A/(1+\tau_A s)$	$10<K_A<400$, $0.02<\tau_A<0.1s$
Exciter	$K_E/(1+\tau_E s)$	$1<K_E<400$, $0.5<\tau_E<1s$
Generator	$K_G/(1+\tau_G s)$	$0.7<K_G<1$, $1<\tau_G<2s$
Sensor	$K_R/(1+\tau_R s)$	$0.001<\tau_R<0.06s$

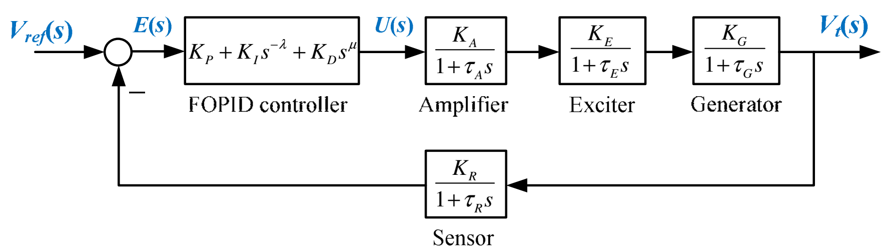


Fig.2. Block diagram of an AVR system with a FOPID controller

2. 3. Multi-objective optimization problems

This subsection presents some basic definitions concerning multi-objective problems [58] used in the following sections.

Definition 2: A multi-objective unconstrained minimization problem with n decision variables and m objectives is generally defined as follows:

$$\text{Minimize } f(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), ..., f_m(\mathbf{x})) \quad (5)$$

where $\mathbf{x}=(x_1, x_2,\ldots, x_n)\in \mathbb{R}^n$ is the vector of n decision variables, each decision variable x_i is bounded with lower and upper limits $l_i \leq x_i \leq u_i$, $i=1, 2, \ldots, n$.

Definition 3: A vector $\mathbf{u}=(u_1, u_2, \ldots, u_m)\in \mathbb{R}^m$ is defined to dominate another vector $\mathbf{v}=(v_1, v_2, \ldots, v_m)\in \mathbb{R}^m$ (denoted as $\mathbf{u} \prec \mathbf{v}$) if and only if all the following two conditions should be satisfied simultaneously: (1) $\forall i \in \{1,2,...,m\}, u_i \leq v_i$, and (2) $\exists i \in \{1,2,...,m\}, u_i < v_i$.

Definition 4: A solution $\mathbf{x} \in \mathbb{R}^n$ is defined to be non-dominated (or Pareto optimal) with respect to $\mathbb{R}^n$ if and only if there does not exist another solution $\mathbf{y} \in \mathbb{R}^n$ such that $f(\mathbf{y}) \prec f(\mathbf{x})$. The Pareto-optimal set consists of all Pareto optimal solutions in the entire search space and the Pareto-optimal front is defined as the set of all objective functions values corresponding to the Pareto optimal solutions.

2.4. Extremal optimization

In general, the τ-EO algorithm, as the original EO version proposed by Boettcher and Percus in the seminal papers [36],[37] and its modified versions consist of the following basic operations, such as initialization of a random solution and the best global fitness so far, evaluation of global fitness and local fitness, selection of some bad local variables randomly or based on power-law probability distribution, generation a new solution by mutation for the selected bad variables, updating the best global fitness so far, and updating the solution by accepting the new solution unconditionally. Clearly, EO algorithms eliminate the bad more than encourage the good [39] because of the above mentioned basic operations including selection of bad local variables, generation a new solution by mutation for the selected bad variables.

3. The proposed algorithm

This section presents the proposed MOEO-based FOPID controller design algorithm. More specifically, definition of multi-objective functions, description of the main algorithm, and analysis of the proposed algorithm are given in the following subsection, respectively.

3.1 Multi-objective functions definition

In order to evaluate the control performance of PID controllers, most of previous research works [1], [6] have used various quantitative indices, such as integral of absolute error (IAE), integral squared error (ISE), integral of time weighted absolute error (ITAE), integral of time multiply squared error (ITSE), etc. In practice, it is difficult to guarantee satisfied performance for a complex system by using a single index [59]. Although some efforts have been devoted to provide more effective optimization methods with weighted fitness by considering several indices [9], [10], it is also not easy to determine appropriate values of various weight parameters for inexperienced designers and engineers. As analyzed in the research work [30], [31], the design of PID and FOPID controllers requires multi-objective optimization because of contradictory objective functions and performance metrics, e.g., ITSE and integral of the squared deviation of controller output (ISDCO). Here, in order to considering the trade-off between transient response and steady-state performance, the design problem with respect to a FOPID controller is formulated as the following multi-objective optimization problem with three objective functions including minimization of IAE, absolute steady-state error, and settling time.

Definition 5: Design of a FOPID controller denoted as $\mathbf{x}=(x_1, x_2, x_3, x_4, x_5)=(K_P, K_I, K_D, \lambda, \mu)$ is defined as an unconstrained multi-objective optimization problem with three contradictory objective functions:

$$\text{Minimize}\, f(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), f_3(\mathbf{x})) \tag{6}$$

$$f_1(\mathbf{x}) = \int_0^{\infty} |e_i(t)| dt \tag{7}$$

$$f_2(\mathbf{x}) = 1000 |E_{ss}| \tag{8}$$

$$f_3(\mathbf{x}) = t_s \tag{9}$$

$$\text{Subject to} \quad \mathbf{L} \le \mathbf{x} \le \mathbf{U} \tag{10}$$

where $e_i(t)$ is i-th system error at the time t, f_1 is IAE, f_2 is absolute steady-state error E_{ss}, weight coefficient 1000 is used to guarantee the scale of coordinate f_2 consistent with other coordinates f_1 and f_3 for better visualization of Pareto front, and f_3 is settling time t_s corresponding to the system output with some level of steady-state error required by designers or engineers. Additionally, $\mathbf{L}= (l_1, l_2, l_3, l_4, l_5)$ and $\mathbf{U}= (u_1, u_2, u_3, u_4, u_5)$, where $l_i \le x_i \le u_i$, i=1, 2, …, 5.

It is should be noted that the selection of multi-objective functions is not only limited to the proposed definition in this work. There are several possible choices based on combination of these above mentioned performance indices for a specified control system, which will be another significant subject of future investigation.

3.2 The main algorithm

In this subsection, we propose MOEO-based FOPID controller design algorithm to solve the multi-objective optimization problem as described by definition 5. This algorithm consists of the following main components, such as generation of a random real-coded individual representing a

FOPID controller, updating the current individual based on polynomial mutation (PLM) [60], Pareto-based fitness assignment strategy based on non-dominated sorting, updating the external archive according to the archive controller and the crowding-distance metric. The flowchart of the proposed algorithm is shown in Fig.3, and more detailed description is given as follows:

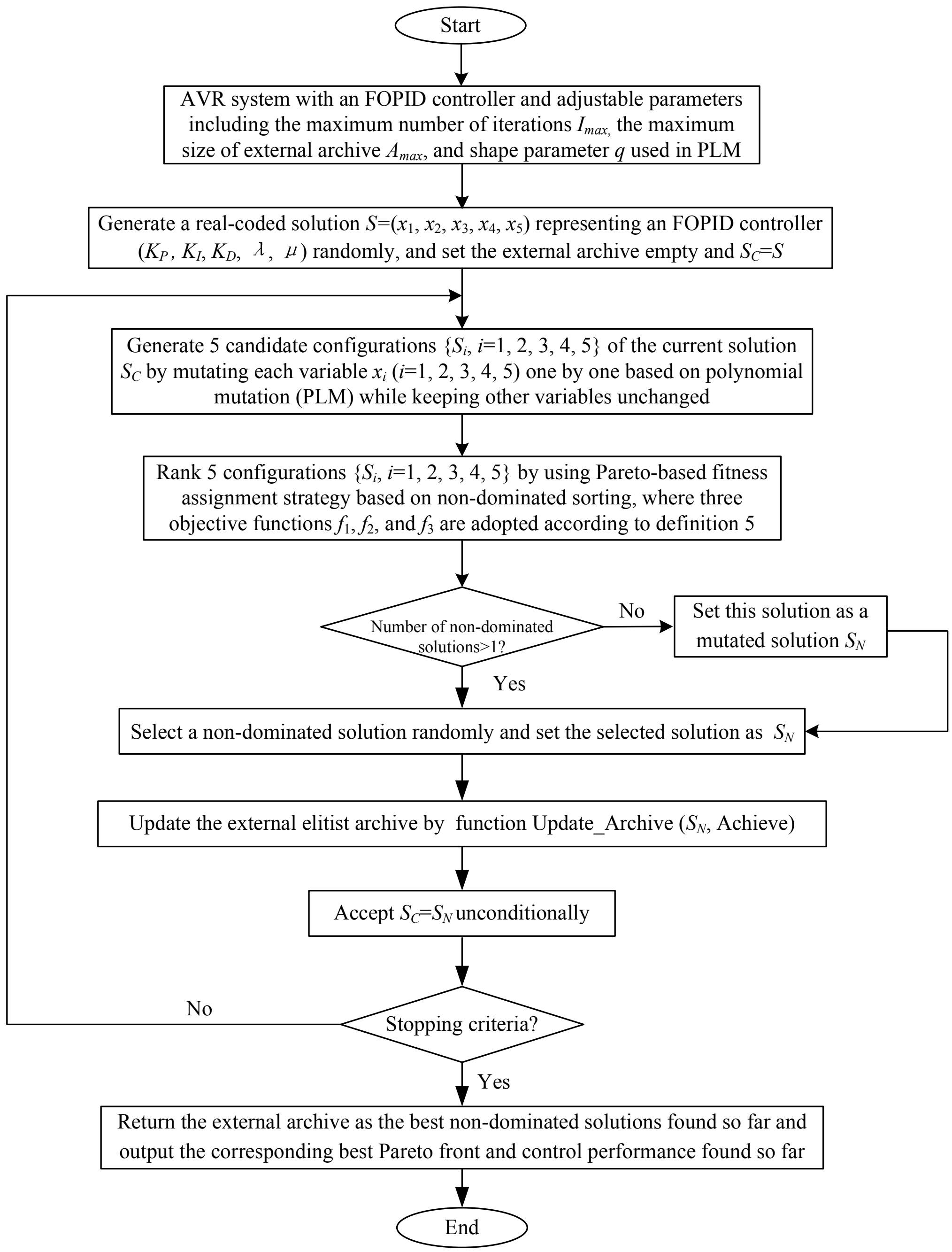


Fig.3 The flowchart of MOEO-based FOPID controller design algorithm

MOEO- FOPID controller design algorithm

Input: AVR system with a FOPID controller and adjustable parameters including the maximum number of iterations $I_{max,}$ the maximum size of external archive A_{max}, and shape parameter q used in

PLM.

Output: The best non-dominated solutions for FOPID controller and the corresponding best Pareto front found so far.

Step 1: Generate a real-coded solution $S=(x_1, x_2, x_3, x_4, x_5)$ representing a FOPID controller (K_P, K_I, K_D, λ, μ) randomly, and set the external archive empty and $S_C=S$.

Step 2: Generate 5 candidate configurations $\{S_i, i=1, 2, 3, 4, 5\}$ of the current solution S_C by mutating each variable x_i (i=1, 2, 3, 4, 5) one by one based on PLM operator [60] while keeping other variables unchanged.

$$S_i(x_i) = S_i(x_i) + \alpha \cdot \beta_{\max}(x_i), i = 1, 2, ..., 5 \tag{11}$$

$$\alpha = \begin{cases} (2r)^{(1/(q+1))} - 1, & \text{if } r < 0.5 \\ 1 - [2(1-r)]^{(1/(q+1))}, & \text{otherwise} \end{cases} \tag{12}$$

$$\beta_{\max}(x_i) = \max[S_i(x_i) - l_i, u_i - S_i(x_i)],\ i = 1, 2, ..., 5 \tag{13}$$

where x_i is the decision variable, q is a positive real number, r is a uniformly distributed random number between 0 and 1, and $\beta_{\max}(x_i)$ is the predefined maximum value of perturbation allowed between original and mutated solution.

Step 3: Rank 5 configurations $\{S_i, i=1, 2, 3, 4, 5\}$ by using Pareto-based fitness assignment strategy based on non-dominated sorting, where three objective functions f_1, f_2, and f_3 are adopted according to definition 5.

Step 4: If there is only non-dominated configuration S_{nd}, then select S_{nd} as the new solution S_N; otherwise, select a non-dominated configuration randomly, and set this selected one as the new solution S_N.

Step 5: Update the external elitist archive by function Update_Archive (S_N, Achieve) shown in Fig.5.

Step 6: Accept $S_C=S_N$ unconditionally.

Step 7: Repeat step 2-6 until the predefined stopping criterion, e.g., maximum number of iterations $I_{\max}$ is met.

Step 8: Return the external archive as the best non-dominated solutions found so far and output the corresponding best Pareto front found so far.

3.3. Pareto-based fitness assignment strategy

In the proposed MOEO-FOPID algorithm, a Pareto-based fitness assignment strategy is introduced based on non-dominated sorting similar to [55], [56]. More specially, the fitness of S is assigned zero when one solution S is non-dominated by all the other solutions. In other words, the non-dominated solutions are ranked as zero while the rank of the worst solutions is the number of all solutions minus one.

To be clearer, the process of Pareto-based fitness assignment strategy based on non-dominated sorting in step 2 for a two-objective minimization problem is illustrated as Fig.4. More specially, assuming that the current solution $S_i=(x_1, x_2, x_3, x_4, x_5)$ marked with black solid circle with 5 decision variables will change to other 5 solutions $S_{iA}=(x_{m1}, x_2, x_3, x_4, x_5)$, $S_{iB}= (x_1, x_{m2}, x_3, x_4, x_5)$, $S_{iC}=(x_1, x_2, x_{m3}, x_4, x_5)$, $S_{iD}=(x_1, x_2, x_3, x_{m4}, x_5)$, $S_{iE}=(x_1, x_2, x_3, x_4, x_{m5})$ via PLM, where x_{m1}, x_{m2}, x_{m3}, x_{m4}, and x_{m5} are the mutated variable from x_1, x_2, x_3, x_4, and x_5 by PLM, respectively. The solution S_{iA} marked red solid circle dominates any of the other four solutions, so the rank number of S_{iA} is 0. Similarly, the rank number of S_{iB}, S_{iC}, S_{iD} marked with green solid circles is 1 because each of S_{iB}, S_{iC}, S_{iD} is non-dominated by S_{iA} while is dominated by the other two solutions from $\{S_{iB}, S_{iC}, S_{iD}\}$. Additionally, the worst case is that the solution $S_{i\mathrm{E}}$ marked with blue solid circle is dominated by any of the other four solutions $S_{i\mathrm{A}}$, $S_{i\mathrm{B}}$, $S_{i\mathrm{C}}$, and $S_{i\mathrm{D}}$, so the fitness of $S_{i\mathrm{E}}$ equals to 4.

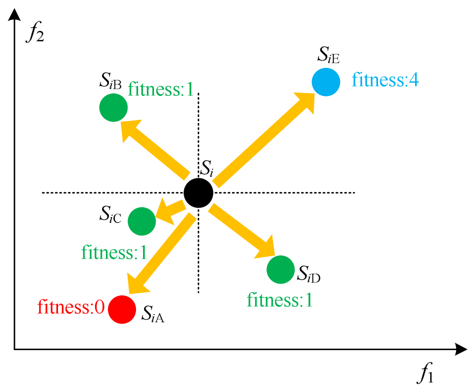


Fig.4. Illustrated process of Pareto-based fitness assignment strategy based on non-dominated sorting

3.4. Update of the external elitist archive

In order to keep a historical record of non-dominated solutions found along the search process of the proposed MOEO-FOPID, the external archive is designed by borrowing the elitist mechanism of MOEO [55] and MOPEO [56]. More specially, the archive consists of two main components: (1) archiving logic [58] designed to decide whether the non-dominated solutions found in the new population should be added to the achieve or not, and (2) crowding-distance metric [32] adopted to judge whether the solutions in the new population reside in the most region of the archive. The detailed description of the external archive is given in Fig.5.

Function Update_Archive(S_{mi}, Archive)

```
Begin function
  If   the solution S_mi is dominated by at least one member of the archive, then
       The archive keeps unchanged
  Else if some members of archive are dominated by S_mi, then
         Remove all the dominated members from the archive and add S_mi to the archive
       End if
  Else
    If the number of archive is smaller than A_max, i.e., the predefined maximum number of the archive, then
      Add S_mi to the archive
    Else
      If   S_mi resides in the most crowded region of the archive, then
           The archive keeps unchanged
      Else
           Replace the member in the most crowded region of the archive by S_mi
      End if
    End if
  End if
End function
```

Fig.5. The pseudo-code of function Update_Archive(S_{mi}, Archive)

3.5. Analysis of the proposed algorithm

In the above description of MOEO-based FOPID controller design algorithm, the adjustable parameters including the maximum number of iterations ($I_{\max}$), the maximum size of external archive A_{max}, and shape parameter q used in PLM play critical roles in controlling the performance of MOEO-FOPID. The comparison of adjustable parameters used in different optimization algorithms-based FOPID and PID controller design algorithms is shown in Table 2. It should be noted that one of most special case is MOEO-PID algorithm when λ=1 and μ=1 in the proposed algorithm. It is clear that the proposed MOEO-FOPID is simpler than NSGA-II-FOPID [30], NSGA-II-PID [30], and other reported single-objective evolutionary algorithms-based FOPID, such as GA-FOPID [14], PSO-FOPID[14],[24], and CAS-FOPID[14], due to its fewer adjustable parameters needing to be tuned. Furthermore, the superiority of the proposed MOEO-FOPID controller to these reported evolutionary algorithms-based FOPID and PID controllers in terms of accuracy and robustness will be demonstrated by a large number of experimental results in the next section.

Table 2: The adjustable parameters used in different evolutionary algorithms-based FOPID and PID controller design algorithms

Algorithm	Number of parameters	Adjustable parameters
GA-FOPID [14]	5	Population size NP, maximum number of iterations $I_{\max}$, select parameter, crossover rate P_c, mutation rate P_m
PSO-FOPID [14],[24]	6	NP, $I_{\max}$, inertia weight factor $w_{\max}$ and $w_{\min}$, acceleration parameter c_1, c_2
CAS-FOPID [14]	6	Number of ants K, $I_{\max}$, sufficient large positive parameter a, parameter $b\in$[0, 2/3], organization factor of the ith ant r_i, initial value of the organization variable $y_i(0)$
NSGA-II-FOPID [30] NSGA-II-PID [30]	9	Number of chromosomes N, number of generation, archive size, tournament size, crossover rate, mutation rate, Pareto front population fraction, initial condition x_0 and parameter a of chaotic map
MOEO-FOPID MOEO-PID	3	maximum number of iterations $I_{\max}$, maximum size of external archive A_{max}, and shape parameter q used in PLM

4. Experimental results

To demonstrate the effectiveness of the proposed MOEO-FOPID algorithm, this section gives the experimental results on AVR system by comparing with NSGA-II-FOPID [30], reported competitive single-objective evolutionary algorithms-based FOPID including GA-FOPID [14], PSO-FOPID [14], and PSO-FOPID [14], NSGA-II-PID[30], and MOEO-PID. For a fair comparison, the parameters of AVR system are set as the same as in the research work [14]: K_A = 10, τ_A = 0.1, K_E = 1, τ_E = 0.4, K_G = 1, τ_G = 1, K_R = 1, and τ_R = 0.01. The lower and upper bounds of each FOPID control parameter are set the same as in [14]: $0\le K_p\le 3$, $0\le K_i\le 1$, $0\le K_d\le 1$, $0\le\lambda\le 2$, $0\le\mu\le 2$, and the sample time T_s is set as 0.01 second. In practice, the approximate optimal value of the adjustable parameters including $I_{\max}$, A_{max} and q in MOEO-FOPID for a specific problem is determined by trial and error. Generally, the larger the values of $I_{\max}$ and A_{max} are, the better the obtained Pareto solutions are, yet the higher the computational cost of the proposed algorithm is. In the experiments implemented in our work, $I_{\max}$ and A_{max} are easily determined by considering the balance between accuracy and computational efficiency. Additionally, the adjustable parameter q is used to control the PLM mutation operation used in MOEO-FOPID algorithm, and it often ranges from 50 to 100. In fact, we have designed and implemented the experiment to study the effect of the adjustable parameter q on the performance of the proposed algorithm. The experimental results have shown that the proposed algorithm is robust when q ranges from 50 to 100. In the following experiments, these adjustable parameters are set as $I_{\max}$=5000, A_{max}=300, and q=90. It should be note that each evolutionary algorithm is executed ten independent runs and all the experiments have been implemented by using MATLAB software based on FOMCON toolbox [61] on a 3.10 GHz PC with processor i5-2400 and 2 GB RAM.

4.1. MOEO-FOPID and its comparison with other evolutionary algorithms-based FOPID

Fig.6 shows the interpolated 3D Pareto front of three objectives f_1, f_2, f_3 for MOEO-FOPID controller, which is constructed for better visualization based on some non-dominated solutions obtained by the proposed MOEO-FOPID algorithm. It is clear that the diversity of these non-dominated solutions is very good. Some representative solutions on 3D Pareto front are presented in Table 3 and the corresponding terminal voltage step response of AVR are shown in Fig.7. Additionally, Table 4 presents some best non-dominated solutions found by MOEO-FOPID algorithm and the corresponding control performance evaluated by overshoot M_p(%), steady-state error E_{ss}, rise time t_r, and settling time t_s with 5% error.

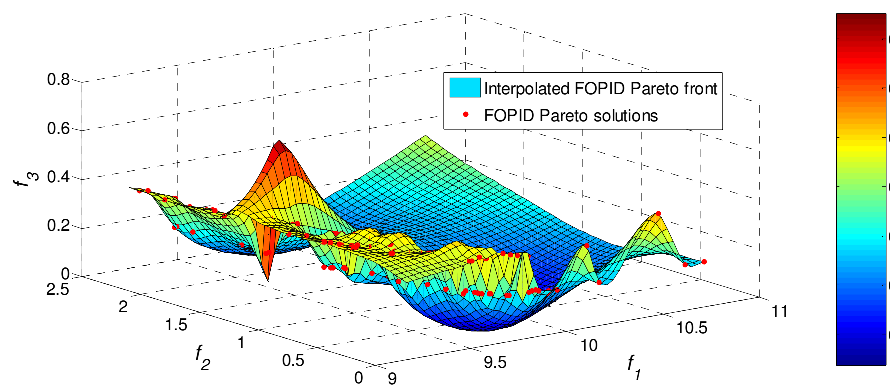


Fig.6. 3D Pareto front of three objectives f_1, f_2, f_3 for MOEO-FOPID controller

Table 3: Representative solutions on the 3D Pareto front of f_1, f_2, and f_3 for MOEO-FOPID controller

Solutions	f_1	f_2	f_3	K_P	K_I	K_D	λ	μ
S_{nd1}	10.16194	0.000923	0.18	2.994396	0.816526	0.520659	1.177759	1.364708
S_{nd2}	9.945877	0.002313	0.18	2.986779	0.876460	0.549036	1.155639	1.345239
S_{nd3}	9.907913	0.011826	0.18	2.949676	0.920846	0.52602	1.139051	1.367165
S_{nd4}	9.908909	0.025028	0.17	2.992812	0.876460	0.542451	1.155639	1.360047
S_{nd5}	9.860025	0.043441	0.17	2.996575	0.895355	0.578998	1.148378	1.336722

Table 5 shows comparative performance of MOEO-FOPID with NSGA-II-FOPID [30] and single-objective evolutionary algorithms-based FOPID controllers, such as GA-FOPID [14], PSO-FOPID [14] CAS-FOPID with β=1 and β=1.5 [14]. Moreover, terminal voltage step response of AVR system with MOEO-FOPID, NSGA-II-FOPID and these aforementioned single-objective evolutionary algorithms- based FOPID controllers are shown in Fig.8. It is evident that MOEO-FOPID provides better performance than NSGA-II-FOPID [30] in terms of all four indices. Although M_p

obtained by MOEO-FOPID is worse than that by PSO-FOPID [14], CAS-FOPID with β=1 and β=1.5 [14], other three performance indices obtained by MOEO-FOPID are all better. Compared with GA-FOPID [14], the proposed MOEO-FOPID provides three better performance indices including M_p(%), t_s and E_{ss}, and only little worse t_r.

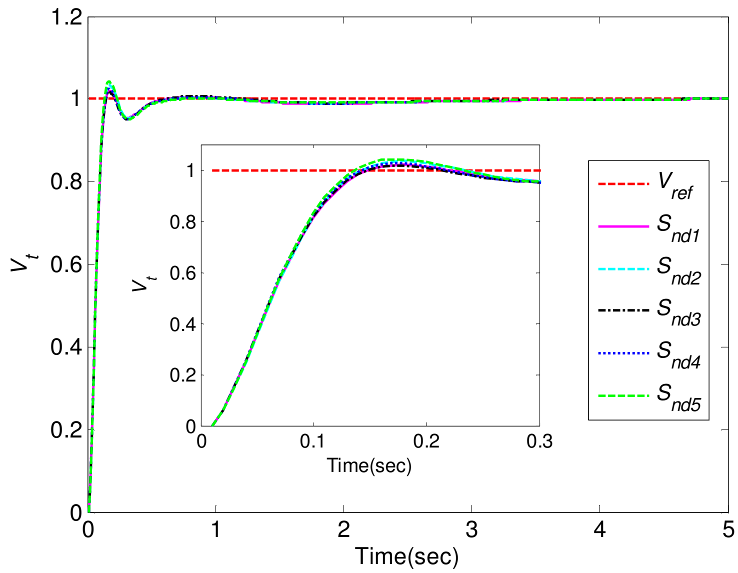


Fig.7. Terminal voltage step response of AVR with representative MOEO-FOPID solutions as reported in Table 3

Table 4: Best non-dominated solutions for MOEO-PID controller and the corresponding control performance

S_{nd}	f_1	f_2	f_3	K_P	K_I	K_D	λ	μ	M_p(%)	t_r	t_s	E_{ss}
1	9.913	6.577E-06	0.18	2.9737	0.9089	0.5383	1.1446	1.3462	3.203	0.13	0.18	6.577E-09
2	10.515	1.618E-05	0.18	2.9620	0.7297	0.5404	1.2096	1.3521	2.758	0.13	0.18	1.618E-08
3	9.825	2.392E-05	0.18	2.8611	0.9701	0.5409	1.1178	1.3444	2.875	0.13	0.18	2.392E-08
4	9.611	2.998E-05	0.17	2.9689	0.9966	0.5695	1.1198	1.3399	3.878	0.13	0.17	2.998E-08
5	9.814	3.097E-05	0.18	2.8501	0.9664	0.5531	1.1174	1.3354	3.479	0.13	0.18	3.097E-08
6	9.882	4.843E-05	0.18	2.8645	0.9496	0.5383	1.1233	1.3425	2.961	0.13	0.18	4.843E-08
7	10.222	4.913E-05	0.18	2.9988	0.7954	0.5588	1.1853	1.3436	3.601	0.13	0.18	4.913E-08
8	10.058	4.994E-05	0.17	2.9948	0.8271	0.5352	1.1736	1.3634	2.210	0.13	0.17	4.994E-08
9	9.967	5.396E-05	0.18	2.8316	0.9214	0.5589	1.1272	1.3267	3.939	0.13	0.18	5.396E-08
10	9.930	6.234E-05	0.18	2.9670	0.9398	0.5543	1.1345	1.3240	4.640	0.13	0.18	6.234E-08

Table 5: Comparative performance of MOEO-FOPID with NSGA-II-FOPID and reported competitive single-objective evolutionary algorithms-based FOPID controllers

Algorithm	K_P	K_I	K_D	λ	μ	M_p(%)	t_r(sec.)	t_s(sec.)	E_{ss}
GA-FOPID [14]	1.6947	0.8849	0.3964	1.0248	1.1296	9.2600	0.1298	0.3395	0.0006
PSO-FOPID [14]	1.6264	0.2956	0.3226	1.3183	1.1980	0.0953	0.1375	0.4563	0.0047
CAS-FOPID (β=1) [14]	1.0537	0.4418	0.2510	1.0624	1.1122	0.1678	0.2223	0.3037	0.0014
CAS-FOPID (β=1.5) [14]	0.9315	0.4776	0.2536	1.0275	1.0838	0.0642	0.2305	0.3187	0.0012
NSGA-II-FOPID [30]	0.8399	1.3359	0.3512	0.9147	0.7107	38.7887	0.3200	1.2700	0.0014

MOEO-FOPID	2.9737	0.9089	0.5383	1.1446	1.3462	3.2038	0.1300	0.1800	6.577E-09

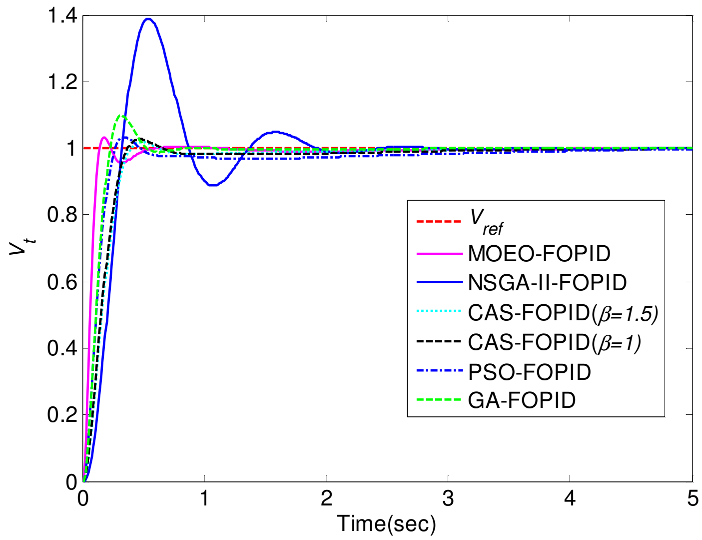


Fig.8. Terminal voltage step response of AVR with MOEO-FOPID, NSGA-II-FOPID and reported competitive single-objective evolutionary algorithms-based FOPID controllers

4.2. MOEO-PID and its comparison with MOEO-FOPID, NSGA-II-FOPID and NSGA-II-PID

As one of the most special case of MOEO-FOPID, the interpolated 3D Pareto front of three objectives f_1, f_2, f_3 for MOEO-PID controller shown in Fig. 9 is similarly constructed for better visualization based on some non-dominated solutions obtained by MOEO-PID algorithm. Clearly, the distribution of these non-dominated solutions for MOEO-PID is also very good. Some representative solutions on 3D Pareto front are presented in Table 6 and the corresponding terminal voltage step response of AVR are shown in Fig.10.

To further demonstrate the effectiveness of the proposed MOEO-FOPID, we give comparative performance of MOEO-FOPID with MOEO-PID, NSGA-II-based FOPID and PID controllers [30] shown in Table 7 and the corresponding terminal voltage step response shown in Fig.11. Although MOEO-PID is worse than MOEO-FOPID, MOEO-PID is superior to NSGA-II-FOPID in terms of all four performance indices. Furthermore, MOEO-PID provides three better performance indices including $M_p(\%)$, t_s, and E_{ss} than NSGA-II-PID yet only little worse t_r. In this sense, the proposed MOEO-FOPID and MOEO-PID are considered as superior to the reported competitive NSGA-II-FOPID [30] and NSGA-II-PID [30].

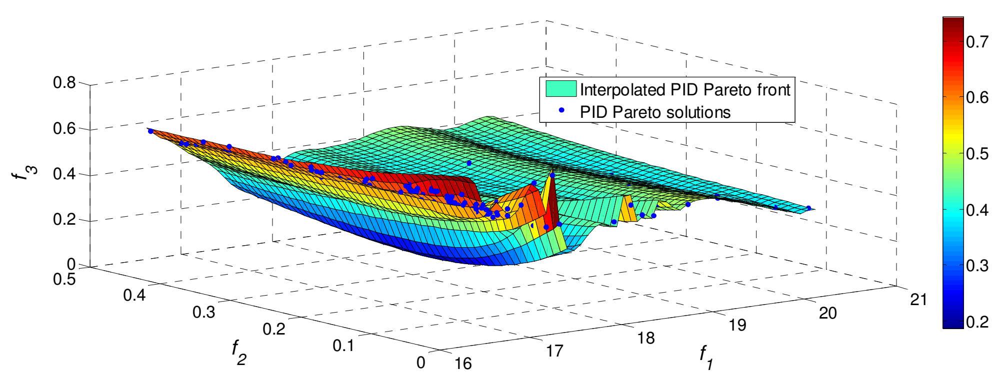


Fig.9. 3D Pareto front of three objectives f_1, f_2, f_3 for MOEO-PID controller

Table 6: Representative solutions on the 3D Pareto front of f_1, f_2, and f_3 for MOEO-PID controller

Solutions	f_1	f_2	f_3	K_P	K_I	K_D	M_p(%)	t_r(sec.)	t_s(sec.)	E_{ss}
S_{nd1}	20.10781	0.009123	0.39	0.85035	0.74732	0.38744	4.50813	0.28	0.39	9.12E-06
S_{nd2}	19.95311	0.036733	0.38	0.86488	0.76205	0.39151	4.90683	0.28	0.38	3.67E-05
S_{nd3}	17.21317	0.006221	0.47	1.47617	0.98968	0.49150	17.08944	0.21	0.47	6.22E-06
S_{nd4}	17.96513	0.008242	0.45	1.14783	0.99160	0.47718	11.47526	0.23	0.45	8.24E-06
S_{nd5}	18.62633	0.039102	0.44	1.01335	0.88620	0.44585	8.58514	0.25	0.44	3.91E-05

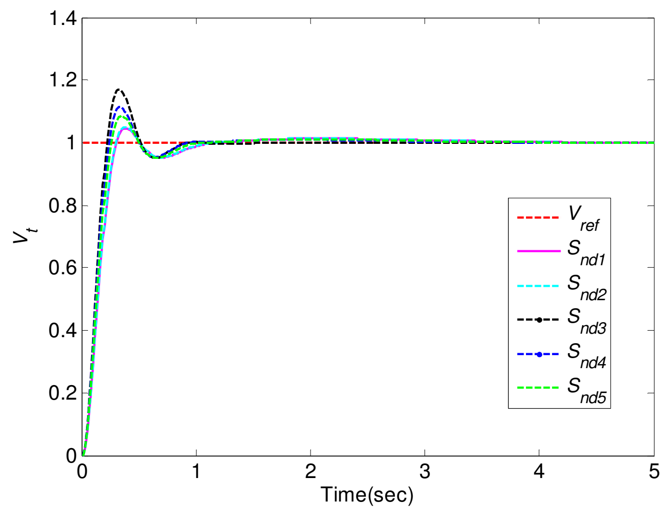


Fig.10. Terminal voltage step response of AVR with representative MOEO-PID solutions as reported in Table 6

Table 7: Comparative performance of MOEO-based with NSGA-II-based FOPID and PID controllers

Algorithm	K_P	K_I	K_D	λ	μ	M_p(%)	t_r(sec.)	t_s(sec.)	E_{ss}
MOEO-FOPID	2.9737	0.9089	0.5383	1.1446	1.3462	3.2038	0.1300	0.1800	6.577E-09
MOEO-PID	0.8503	0.7473	0.3874	1.0000	1.0000	4.5081	0.2800	0.3900	9.123E-06
NSGA-II-FOPID [30]	0.8400	1.3359	0.3512	0.9147	0.7107	38.7887	0.3200	1.2700	1.390E-03
NSGA-II-PID [30]	12.1027	6.0673	7.7007	1.0000	1.0000	63.0904	0.2100	0.5100	4.864E-05

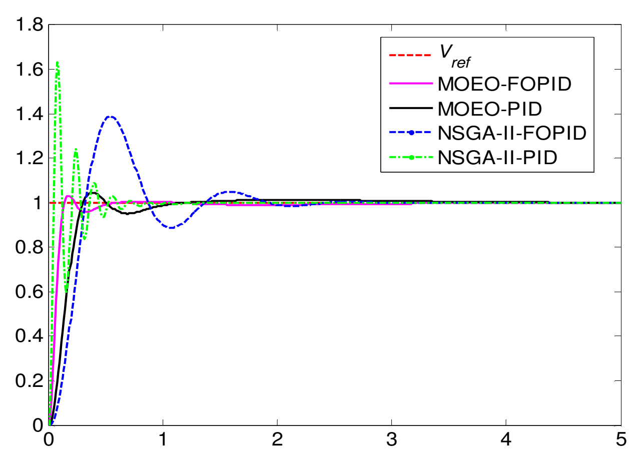


Fig.11. Terminal voltage step response of AVR with MOEO-based with NSGA-II-based FOPID and PID controllers

4.3. Robustness test

To illustrate the robustness of MOEO-FOPID controller, the following experiments considering parameter uncertainties in AVR system due to the change in load condition are implemented.

4.3. 1.Generator uncertainty

When the parameter K_G changes to 0.8 from actual value 1 and τ_G changes to 1.6 from actual value 1 due to the change in load condition, the comparative performance of MOEO-FOPID controller with NSGA-II-based FOPID controller [30] and other single-objective evolutionary algorithms-based FOPID controllers [14] are shown in Table 8 and the corresponding terminal voltage step response are shown in Fig.12. Clearly, the proposed MOEO-FOPID controller is more robust and better than other evolutionary algorithms-based FOPID controllers [14] in terms of at least three performance indices under the uncertainty of the generator. Moreover, Fig.13 presents the terminal voltage step response for AVR system with MOEO-FOPID controller when K_G varied from 1.0 to 0.9, 0.8, 0.7 and τ_G from 1.0 to 1.3, 1.7, 1.9, 2.0, respectively. It is obvious that MOEO-FOPID controller is robust under the variation of parameters K_G and τ_G in the range as given in Table 1.

Table 8: Comparative performance of different evolutionary algorithms-based FOPID controller when K_G changes to 0.8 from 1 and τ_G changes to 1.6 from actual value 1

Algorithm	M_p(%)	t_r(sec.)	t_s(sec.)	E_{ss}
GA-FOPID [14]	5.803333	0.38	0.82	0.0022
PSO-FOPID [14]	1.062411	0.47	0.76	0.0030
CAS-FOPID(β=1) [14]	2.878105	0.61	1.04	0.0026
CAS-FOPID(β=1.5) [14]	2.895868	0.65	1.74	0.0021
NSGA-II-FOPID [30]	32.84993	0.47	1.36	0.0016
MOEO-FOPID	3.264274	0.32	0.72	0.0016

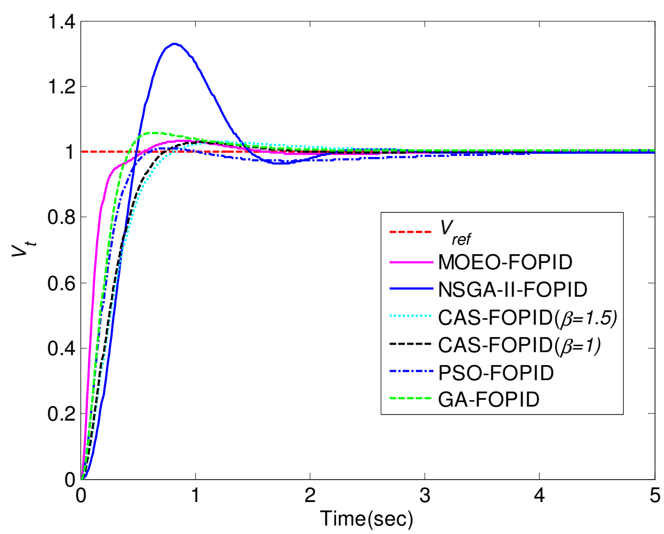


Fig.12. Comparison of terminal voltage step response when K_G changes to 0.8 from 1 and τ_G changes to 1.6 from 1

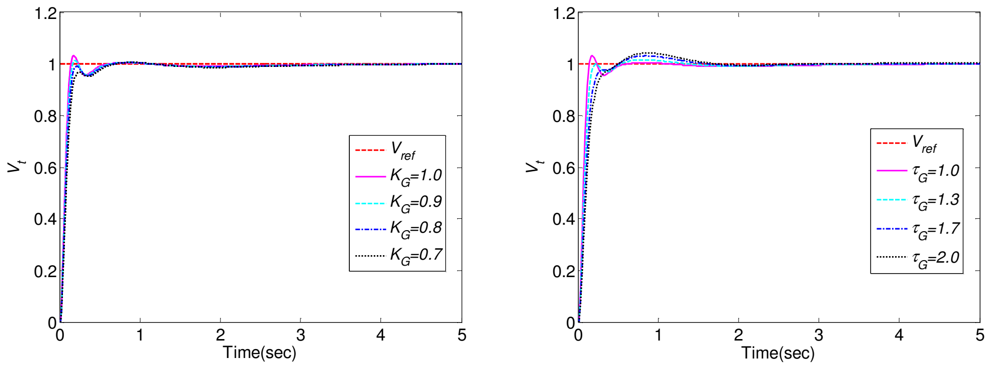


Fig.13. The terminal voltage step response for AVR system with MOEO-FOPID controller under variation of K_G (left) and τ_G (right)

4.3.2. Exciter uncertainty

Table 9: Comparative performance of different evolutionary algorithms-based FOPID controller when K_E changes to 1.2 from 1 and τ_E changes to 0.5 from actual value 0.4.

Algorithm	M_p(%)	t_r(sec.)	t_s(sec.)	E_{ss}
GA-FOPID [14]	13.32531	0.21	0.49	7.12E-04
PSO-FOPID [14]	6.749589	0.23	0.43	5.82E-04
CAS-FOPID(β=1) [14]	7.089991	0.31	0.61	3.13E-04
CAS-FOPID(β=1.5) [14]	6.248848	0.33	0.73	2.59E-04
NSGA-II-FOPID [30]	45.38916	0.32	1.76	9.88E-04
MOEO-FOPID	4.95083	0.13	0.19	1.57E-04

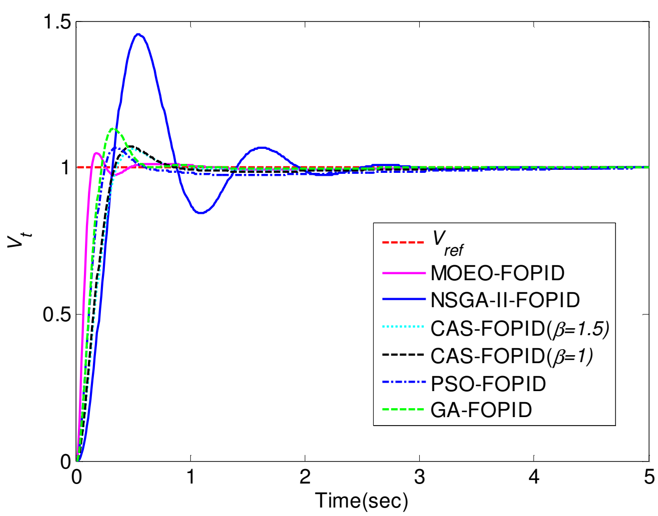


Fig.14. Terminal voltage step response when K_E changes to 1.2 from 1 and τ_E changes to 0.5 from actual value 0.4

Assuming the parameter K_E changes to 1.2 from actual value 1 and τ_E changes to 0.5 from actual value 0.4 due to the change in load condition, Table 9 shows the comparative performance of MOEO-FOPID controller with NSGA-II-based FOPID controller [30] and other single-objective evolutionary algorithms-based FOPID controllers [14], and Fig.14 presents the corresponding terminal voltage step response. It is evident that the proposed MOEO-FOPID controller provides more robust and better performance than NSGA-II-FOPID [30] and competitive single-objective evolutionary algorithms-based FOPID controllers [14] in terms of all four indices under the uncertainty of the exciter. Additionally, when K_G varied from 1.0 to 1.2, 1.5, 1.8, 2.0 and τ_G from 0.4 to 0.5, 0.7, 0.9, 1.0, the terminal voltage step response for AVR system with MOEO-FOPID controller is shown in Fig.15. Obviously, MOEO-FOPID controller is robust under the variation of parameters K_E and τ_E in the range as given in Table 1.

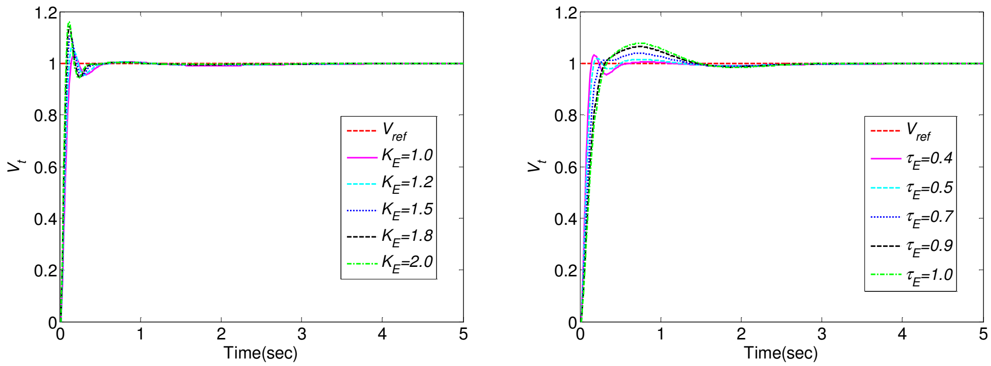


Fig.15. Terminal voltage step response for AVR system with MOEO-FOPID controller under variation of K_E (left) and τ_E (right)

4.3.3. Amplifier uncertainty

Here, the uncertainty of amplifier model parameters is considered, for example, K_A changes to 20 from actual value 10 and τ_A changes to 0.07 from actual value 0.1. The comparative performance of the proposed MOEO-FOPID with NSGA-II-FOPID and competitive single-objective evolutionary algorithms-based FOPID controllers and the corresponding terminal voltage step response of AVR system are presented in Table 10 and Fig. 16, respectively. It is clear that the proposed MOEO-FOPID is also more robust and better than these reported NSGA-II-FOPID, GA-FOPID [14], PSO-FOPID [14], CAS-FOPID with β=1 and β=1.5 [14] under the uncertainty of amplifier model parameters. Furthermore, Fig.17 presents the terminal voltage step response for AVR system with MOEO-FOPID controller when K_A varied from 10 to 30, 50, 100, 200, 400 and τ_A from 1.0 to 1.3, 1.7, 1.9, 2.0, respectively. Clearly, as the value of parameter K_A increases, the overshoot M_p increases but rising time t_r and settling time t_s becomes faster and the steady-state error E_{ss} is smaller. As the value of parameter τ_A increases, M_p becomes smaller while t_r, t_s, and E_{ss} are larger. However, from the perspective of engineering design and system operation, the performance of MOEO-FOPID is accepted by engineers under the variation of parameters K_A and τ_A in the range as given in Table 1. In this sense, the developed MOEO-FOPID controller is viewed as robust for the uncertainty of amplifier model parameters within the range defined as in Table 1.

Table 10: Comparative performance of different evolutionary algorithms-based FOPID controller when K_A changes to 20 from 10 and τ_A changes to 0.7 from actual value 1

Algorithm	M_p(%)	t_r(sec.)	t_s(sec.)	E_{ss}
GA-FOPID [14]	17.27371	0.12	0.26	4.05E-04
PSO-FOPID [14]	10.46691	0.12	0.24	4.63E-04
CAS-FOPID(β=1) [14]	9.079521	0.17	0.32	2.35E-04
CAS-FOPID(β=1.5) [14]	8.941498	0.17	0.36	1.89E-04
NSGA-II-FOPID [30]	46.71605	0.2	1.12	6.92E-04
MOEO-FOPID	14.07733	0.07	0.23	2.98E-05

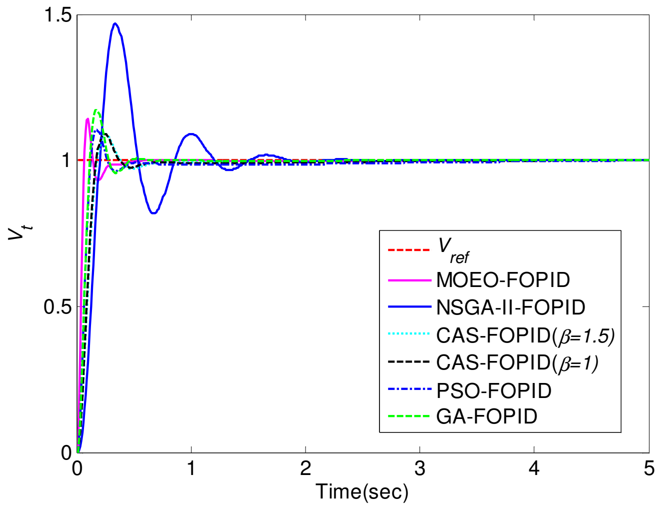


Fig.16. Terminal voltage step response when K_A changes to 20 from 10 and τ_E changes to 0.07 from 0.1

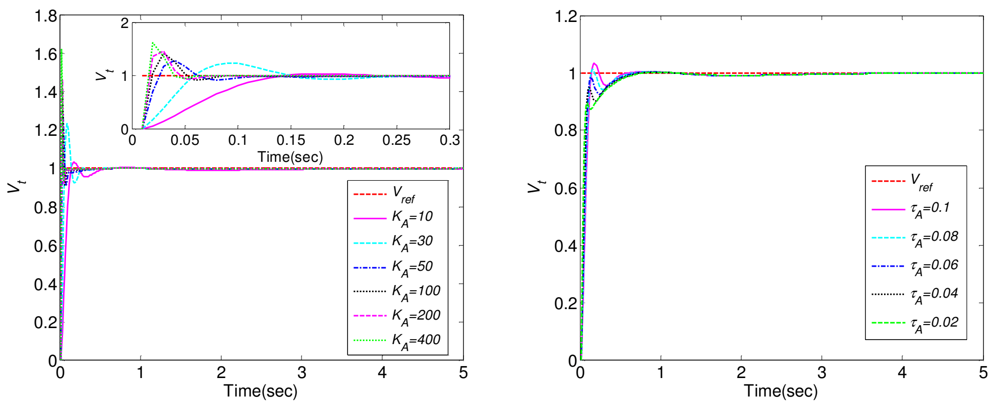


Fig.17. The terminal voltage step response for AVR system with MOEO-FOPID controller under variation of K_A (left) and τ_A (right)

5. Conclusion

From the perspective of multi-objective optimization, a novel FOPID controller design method called MOEO-FOPID for AVR system is proposed based on multi-objective extremal optimization (MOEO) in this work. Firstly, the FOPID design problem for AVR system is formulated as a multi-objective optimization problem with three objective functions including integral of absolute error (IAE), absolute steady-state error, and settling time, and then this problem is solved by developing an improved MOEO algorithm on the basis of individual-based iterated optimization mechanism and polynomial mutation (PLM). One of the most attractive advantages is the relative simplicity of MOEO-FOPID comparing with chaotic NSGA-II-FOPID [30], NSGA-II-PID [30], and single-objective evolutionary algorithms-based FOPID algorithms such as GA-FOPID[14], PSO-FOPID [14],[24], and CAS-FOPID [14] due to its fewer adjustable parameters and single individual-based iterated optimization mechanism with only mutation operation. Furthermore, extensive experimental results have shown that the proposed MOEO-FOPID algorithm provides better or at least competitive performance than these aforementioned evolutionary algorithms in terms of accuracy and robustness. Consequently, the proposed MOEO-FOPID is considered as another novel promising multi-objective evolutionary algorithm to design FOPID controllers for AVR and other industrial systems. However, the performances of MOEO-FOPID may be further improved by tuning the adjustable parameters based on an adaptive mechanism and using other possible appropriate definition of multi-objective functions. On the other hand, the extension of MOEO-FOPID to more complex practical control systems will be another significant subject of future investigation.

Acknowledgements

The authors would like to thank the anonymous referees for their many useful comments and constructive suggestions. This work was partially supported by National Natural Science Foundation of China (Nos. 51207112, 61005049), Zhejiang Province Science and Technology Planning Project (No.2014C31074, 2014C31093), Zhejiang Provincial Natural Science Foundation of China (No. LQ14F030007), and the Program of Xinmiao (Potential) Talents in Zhejiang Province (No. 2014R424014).

References

[1] H. Zhang, J. Wang, Combined feedback–feedforward tracking control for networked control systems with probabilistic delays, J. Franklin Inst. 351 (6) (2014) 3477-3489.

[2] H. Zhang, X. Zhang, J. Wang, Robust gain-scheduling energy-to-peak control of vehicle lateral dynamics stabilisation, Vehicle Syst. Dyn. 52(3) (2014) 309-340.

[3] H. Zhang, Y. Shi, J. Wang, On energy-to-peak filtering for nonuniformly sampled nonlinear systems: a Markovian jump system approach, IEEE Trans. Fuzzy Syst. 22 (1) (2014) 212-222.

[4] H. Zhang and J. Wang. State estimation of discrete-time Takagi-Sugeno fuzzy systems in a network environment, IEEE Trans. Cybern., 2014 (In press)

[5] K. J. Åström, T. Hägglund, The future of PID control, Control Eng. Pract. 9(11) (2001) 1163-1175.

[6] K. H. Ang, G. Chong, Y. Li, PID control system analysis, design and technology, IEEE Trans. Control Syst. Technol. 13(4) (2005) 559-576.

[7] J. Ye, Adaptive control of nonlinear PID-based analog neural networks for a nonholonomic mobile robot, Neurocomputing 71 (2008) 1561-1565.

[8] J. Kang, W. J. Meng, A. Abraham, H. B. Liu, An adaptive PID neural network for complex nonlinear system, Neurocomputing 135 (2014) 79-85.

[9] G. Q. Zeng, K. D. Lu, Y.X. Dai, Z. J. Zhang, M. R. Chen, C. W. Zheng, D. Wu, W.W. Peng, Binary-coded extremal optimization for the design of PID controllers, Neurocomputing 138 (2014) 180-188.

[10] G. Q. Zeng, J. Chen, M. R. Chen, Y. X. Dai, L. M. Li, K. D. Lu, C.W. Zheng, Design of multivariable PID controllers using real-coded population-based extremal optimization, Neurocomputing, 151 (2015) 1343-1353.

[11] I. Podlubny, Fractional-order systems and $PI^{\lambda}D^{\mu}$ controllers, IEEE Trans. Automat. Contr. 44(1) (1999) 208-213.

[12] Y. Q. Chen, I. Petráš, D.Y. Xue, Fractional order control-A tutorial, in: Procedings of 2009 American Control Conference, 2009, pp.1397-1411.

[13] 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.

[14] Y. G. Tang, M. Y. Cui, C. C. Hua, L. X. Li, Y. X.Yang, Optimum design of fractional order $PI^{\lambda}D^{\mu}$ controller for AVR system using chaotic ant swarm, Expert Syst. Appl. 39 (2012) 6887-6896.

[15] H. S. Li, Y. Luo, Y. Q. Chen, A Fractional Order Proportional and Derivative (FOPD) Motion Controller: Tuning Rule and Experiments, IEEE Trans. Control Syst. Technol.18 (2) (2010) 516-520.

[16] A. J. Calderón, B. M. Vinagre, V. Feliu, Fractional Order Control Strategies for Power Electronic Buck Converters, Signal Process. 86 (2006) 2803-2819.

[17] C. A. Monjea, B. M. Vinagreb, V. Feliuc, Y. Q. Chen, Tuning and auto-tuning of fractional order controllers for industry applications, Control Eng. Pract. 16 (2008) 798-812.

[18] J. Cervera, A. Banos, C. Monje, and B. Vinagre, Tuning of fractional PID controllers by using QFT, in: Proceedings of 32nd annual conference on IEEE Industrial Electronics, 2006, pp. 5402-5407.

[19] S. E. Hamamci, An algorithm for stabilization of fractional-order time delay systems using fractional-order PID controllers, IEEE Trans. Automat. Contr. 52 (10) (2007) 1964-1969.

[20] M. K. Bouafoura, N. B. Braiek, $PI^{\lambda}D^{\mu}$ controller design for integer and fractional plants using piecewise orthogonal functions, Commun. Nonlinear Sci. 15 (2010) 1267-1278.

[21] F. Padula, A.Visioli, Tuning rules for optimal PID and fractional-order PID controllers, J. Process Contr. 21 (2011) 69-81.

[22] Z. Gao, M. Yan, J. X. Wei, Robust stabilizing regions of fractional-order PD^{μ} controllers of time-delay fractional-order systems, J. Process Contr. 24 (2014) 37-47.

[23] 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 Eng. Pract. 17 (2009)1380-1387.

[24] H. Remezanian, S. Balochian, A. Zare, Design of optimal fractional-order PID controllers using particle swarm optimization algorithm for automatic voltage regulator (AVR) system, Journal of Control Automation and Electrical Systems 24 (5) (2013) 601-611.

[25] A. Biswas, S. Das, A. Abraham, S. Dasgupta, Design of fractional-order $PI^{\lambda}D^{\mu}$ controllers with an improved differential evolution, Eng. Appl. Artif. Intel. 22 (2009) 343-350.

[26] H. Gozde, M. C. Taplamacioglu, Comparative performance analysis of artificial bee colony algorithm for automatic voltage regulator (AVR) system, J. Franklin Inst. 348 (2011) 1927-1946.

[27] C. H. Lee, F. K. Chang, Fractional-order PID controller optimization via improved electromagnetism-like algorithm, Expert Syst. Appl. 37 (2010) 8871-8878.

[28] K. Tehrani, A. Amirahmadi, S. Rafiei, G. Griva, L. Barrandon, M. Hamzaoui, I. Rasoanarivo, F. Sargos, Design of fractional order PID controller for boost converter based on multi-objective optimization, in: Proceedings of 14^{th} International power electronics and motion control conference (EPE/PEMC), 2010, pp.179-185.

[29] L. Meng, D. Y. Xue, Design of an optimal fractional-order PID controller using multi-objective GA optimization, in: Proceedings of Chinese Control and Decision Conference, 2009, pp.3849-3853.

[30] I. Pan, S. Das, Chaotic multi-objective optimization based design of fractional order $PI^{\lambda}D^{\mu}$ controller in AVR system, Int. J. Elec. Power 43 (2012) 393-407.

[31] I. Pan, S. Das, Frequency domain design of fractional order PID controller for AVR system using chaotic multi-objective optimization, Int. J. Elec. Power 51 (2013) 106-118.

[32] K. Deb, A. Pratab, S. Agrawal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Trans. Evolut. Comput. 6 (2) (2002) 182-197.

[33] Z. L.Gaing, A particle swarm optimization approach for optimum design of PID controller in AVR System, IEEE Trans. Energy Conver. 19 (2) (2004)384-391.

[34] P. Bak, C. Tang, K. Wiesenfeld, Self-Organized Criticality, Phys. Rev. Lett. 59 (1987) 381-384.

[35] P. Bak, K. Sneppen, Punctuated equilibrium and criticality in a simple model of evolution, Phys. Rev. Lett. 71(24) (1993) 4083-4086.

[36] S. Boettcher, A. G. Percus, Extremal optimization: methods derived from co-evolution, in: Proceedings of Genetic and Evolutionary Computation Conference, Orlando, Florida, 1999, pp. 825-832.

[37] S. Boettcher, A.G. Percus., Nature's way of optimizing, Artif. Intell. 119 (2000) 275-286.

[38] S. Boettcher, A.G. Percus, Optimization with extremal dynamics, Phys. Rev. Lett. 86(23) (2001) 5211-5214.

[39] S. Boettcher, M. Grigni, Jamming model for the extremal optimization heuristic, J. Phys. A 35 (5) (2002) 1109-1123.

[40] Y. Z. Lu, M. R. Chen, Y. W. Chen, Studies on extremal optimization and its applications in solving real world optimization problems, in: Proceedings of the 2007 IEEE Symposium on Foundations of Computational Intelligence, Hawaii, USA, 2007, pp.162-168.

[41] S. Boettcher, A.G. Percus, Extremal optimization for graph partitioning, Phys. Rev. E 64(2) (2001) 026114.

[42] S. Boettcher, A.G. Percus, Extremal optimization at the phase transition for the three-coloring problem, Phys. Rev. E 69 (2004) 066703.

[43] Y. W. Chen, Y. Z. Lu, P. Chen, Optimization with extremal dynamics for the travelling salesman problem, Physica A 385 (2007)115-123.

[44] G. Q. Zeng, Y.Z. Lu, W. J. Mao, Multistage extremal optimization for hard travelling salesman problem, Physica A 389 (2010) 5037-5044.

[45] M. B. Menaï, M. Batouche, An effective heuristic algorithm for the maximum satisfiability problem, Appl. Intell. 24 (2006) 227-239.

[46] G. Q. Zeng, Y. Z. Lu, W. J. Mao, Modified extremal optimization for the maximum satisfiability problem, J. Zhejiang Univ.-Sci. C 12(7) (2011) 589-596.

[47] M. R. Chen, X. Li, X. Zhang, Y. Z. Lu, A novel particle swarm optimizer hybridized with extremal optimization, Appli. Soft Comput.10(2010)367-373.

[48] J. Duch, A. Arenas, Community detection in complex networks using extremal optimization, Phys. Rev. E 72 (2) (2005) 027104.

[49] Y. W. Chen, Y. Z. Lu, G. K. Yang, Hybrid evolutionary algorithm with marriage of genetic algorithm and extremal optimization for production scheduling, Int. J. Adv. Manuf. Technol. 36(9-10) (2008) 959-968.

[50] F. L. Sousa, V. Vlassov, F. M. Ramos, Heat pipe design through generalized extremal optimization, Heat Transf. Eng., 25(7) (2004)34-45.

[51] J. Ding, Y. Z. Lu, J. Chu, Extremal optimization for unit commitment problem for power systems, in: Proceedings of IEEE Power and Energy Society General Meeting, 2012, pp.1-8.

[52] S. Boettcher, Evolutionary dynamics of extremal optimization, Lect. Notes Comput. Sci. 5581 (2009) 1-14.

[53] G. Q. Zeng, Y. Z. Lu, Survey on computational complexity with phase transitions and extremal optimization, in: Proceedings of 48th IEEE conference on Control and Decision&28th Chinese Control Conference, 2009, pp.4352-4359.

[54] P. G. Menese, M. Randall, A. Lewis, A hybrid multi-objective extremal optimization approach for multi-objective cominatorial optimization problems, in: Proceedings of IEEE Congress on Evolutionary Computation (CEC), 2010, pp.1-8.

[55] M. R. Chen, Y. Z. Lu, A novel elitist multiobjective optimization algorithm: multiobjective extremal optimization, Eur. J. Oper. Res. 188(3) (2008) 637-651.

[56] M. R. Chen, Y. Z. Lu, G. K. Yang, Multiobjective optimization using population-based extremal optimization, Neural Comput. Appl. 17(2) (2008)101-109.

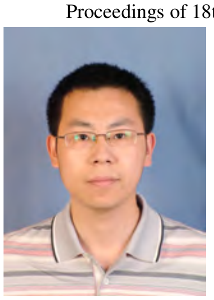
[57] I. Podlubny, Fractional Differential Equations, Academic Press, San Diego, 1999.

[58] C.A.C. Coello, G.T. Pulido, M. S. Leehuga, Handling multiple objectives with particle swarm optimization, IEEE Trans. Evolut. Comput. 8 (3) (2004) 256-279.

[59] J. H. Zhang, J. Zhuang, H. F. Du, S. A. Wang, Self-organizing genetic algorithm based tuning of PID controllers, Inform. Sciences 179(2009) 1007-1018.

[60] K. Deb, M. Goyal, A combined genetic adaptive search (GeneAS) for engineering design, Journal of Computer Science and Informatics, 26 (4) (1996) 30-45.

[61] A. Tepljakov, E. Petlenkov, J. Belikov, FOMCON: Fractional-Order Modeling and Control Toolbox for MATLAB, in: Proceedings of 18th International Conference Mixed Design of Integrated Circuits and Systems, 2011, pp.684-689.



Guo-Qiang Zeng received the PhD degree in Control Science and Engineering from Zhejiang University, China, in 2011. He is currently associate professor at Department of Electrical and Electronic Engineering, Wenzhou University, China. His research interests include computational intelligence, intelligent control, mircogrid, and power electronics.

Jie Chen is currently an undergraduate at Department of Electrical and Electronic Engineering, Wenzhou University, China. His research interests include evolutionary computation and intelligent control.

Yu-Xing Dai received his PhD degree in Control Theories and Control Engineering from Central South University, China, in 2003. He is currently professor at Wenzhou University, China. His research interests include power electronics, computer numerically controlled machine tools, and mircogrid.

Li-min Li received his Ph.D. degree in communication and information systems from Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences, China, in 2012. He is currently a lecturer at Wenzhou University, China. His current research interests include intelligent information processing, GPS integrity, GNSS modernization, and GNSS receiver architectures.

Chong-Wei Zheng is currently professor at Wenzhou University, China. His research interests include power electronics and photoelectric information processing.

Min-Rong Chen received her PhD degree from the Department of Automation, Shanghai Jiao Tong University, China, in 2008. She is currently associate professor at the College of Information Engineering, Shenzhen University, China. Her research interests include evolutionary computation and multi-objective optimization.