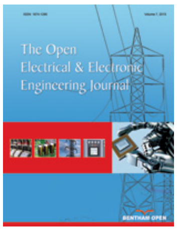
The Open Electrical & Electronic Engineering Journal

Content list available at: www.benthamopen.com/TOEEJ/

DOI: 10.2174/1874129001610010101



RESEARCH ARTICLE

PID Parameters Optimization Research for Hydro Turbine Governor by an Improved Fuzzy Particle Swarm Optimization Algorithm

Chen Gonggui[1,2,*], Du Yangwei[2], Guo Yanyan[3], Huang Shanwai[2] and Liu Lilan[2]

[1] *Key Laboratory of Industrial Wireless Network and Networked Control of Ministry of Education, Chongqing University of Posts and Telecommunications, Chongqing 400065, China*

[2] *Research Center on Complex Power System Analysis and Control, Chongqing University of Posts and Telecommunications, Chongqing, 400065, China*

[3] *The Department of Locomotive and Vehicle Engineering, Wuhan Railway Vocational College of Technology, Wuhan 430205, China*

Received: April 08, 2016 Revised: August 22, 2016 Accepted: August 23, 2016

Abstract: Parameter optimization of water turbine regulating system (WTRS) is decisive in providing support for the power quality and stability analysis of power system. In this paper, an improved fuzzy particle swarm optimization (IFPSO) algorithm is proposed and used to solve the optimization problem for WTRS under frequency and load disturbances conditions. The novel algorithm which is based on the standard particle swarm optimization (PSO) algorithm can speed up the convergence speed and improve convergence precision with combination of the fuzzy control thought and the crossover thought in genetic algorithm (GA). The fuzzy control is employed to get better dynamics of balance between global and local search capabilities, and the crossover operator is introduced to enhance the diversity of particles. Two different types of WTRS systems are built and analyzed in the simulation experiments. Furthermore, the sum of regulating time and another number that is the integral of sum for absolute value of system error and the squared governor output signal is considered as the fitness function of this algorithm. The simulation experiments for parameter optimization problem of WTRS system are carried out to confirm the validity and superiority of the proposed IFPSO, as compared to standard PSO, Ziegler Nichols (ZN) algorithm and fuzzy PID algorithm in terms of parameter optimization accuracy and convergence speed. The simulation results reveal that IFPSO significantly improves the dynamic performance of system under all of the running conditions.

Keywords: Crossover thought, Dynamic performance, Fuzzy control thought, Improved fuzzy particle swarm optimization (IFPSO), Water turbine regulating system.

1. INTRODUCTION

Hydropower is a kind of renewable and pollution-free clean energy with the advantages of low resources consumption and widespread distribution; besides, it does much less harm to the environment in comparison with thermal power. The electrical safety and quality are primary considerations for hydropower station operation, which strongly depend on the stability of water turbine regulating system (WTRS) [1, 2]. What's more, governor parameters play a crucial role in the performance indicators of WTRS system. As the dominant regulation rule of water turbine governor [3, 4], PID regulation gets widely applied in engineering practice due to its practicability and robustness, and the most important problem with respect to this regulate law is how to optimally tune the parameters to obtain satisfactory control performance. Over the past decades, in order to improve the tuning performance, a great variety of optimization algorithms have been put forward and successfully applied to the adjustment of PID parameters, which can

* Address correspondence to this author at the Chongqing University of Posts and Telecommunications, Chongqing, P.R. China; Tel: +86 15696106539; E-mail:chenggpower@126.com

be divided into two main categories: traditional parameter setting methods and intelligent algorithms.

In [5], the principle of root locus was applied to determine the optimal value of parameters in a soft feedback type of governor according to the requirements of pole assignment. In literature [6], the author introduced a kind of control parameter optimization method based on orthogonal test, and this method was successfully implemented in a synthesis test instrument and integrated into a microcomputer-based hydro-turbine governor. In [7], taking the stability of system as the primary purpose, a strategy was proposed to set turbine governor parameters, which determined K_P in accordance with the given frequency to guarantee the stability of system and curb excessive amount of overshoot produced by frequency response; besides, K_I and K_D were determined through the disturbances rejection optimization model. And then, the optimal control for two modes can be achieved to the maximum extent.

Genetic algorithm was used to optimize parameters of turbine governor [8], and for all cases, the experiment results took on a good dynamic performance of system. In literature [9], differential evolution algorithm was used for the hydraulic servo control system PID parameters tuning, whose simulation results showed that it was an effective strategy to make the performance of system good. Proposed in [10], an improved algorithm which was based on gravitational search algorithm was combined with the particles' memory characteristic and the way of information sharing in PSO algorithm to optimize the PID parameters for an excitation control system.

Compared with intelligent algorithms, traditional PID parameters setting methods are prone to oscillation and the large amount of overshoot when deal with some complex systems. What's worth mentioning is that the superiority of particle swarm optimization algorithm is prominent in numerous intelligent algorithms. It was proposed as a kind of swarm intelligence optimization algorithm by Kennedy and Eberhart in 1995, which was inspired by social behavior of animals [11]. Compared with some other optimization algorithms, PSO can be implemented through simple programming, and especially, it often can find the better global optimal value [12 - 15]. Over the past few decades, PSO algorithm has been successfully applied in many practical engineering problems and got great development.

PSO algorithm converges rapidly in the initial stage of the search, but always slows down considerably in the late stage and suffers the problem of being trapped in the local optima [16]. In order to overcome these shortcomings, numerous variants have been introduced and reported. The local mean value concept was introduced into PSO to increase the particles diversity and avoid algorithm trapping into local optima [17]. Xia *et al*. [18] put forward three kinds of strategies to improve the efficiency of algorithm: tabu detecting, shrinking and local learning strategies. Wei *et al*. [19] introduced the choice thought in genetic algorithm into PSO, in which particles with lower fitness were replaced by those with higher fitness so as to accelerate the algorithm convergence speed. In [20], chaos optimization movement and crossover thought from the difference evolution algorithm were introduced into PSO to respectively enhance local exploitation ability and the diversity of particles.

It must be pointed out that the inertia weight, playing an important role in velocity update of particles during the process of iteration, dominates the balance between the exploration capability in global scope and the exploitation capability in the local scope for particles. The PSO searching process is a complex and nonlinear process, and in the standard PSO, the linear decreasing inertia weight strategy only can obtain a linear transition of search capability from global to local search but does not really reflect the actual searching process of finding the optimal value.

In order to improve the performance of standard PSO, the inertia weight should be dynamically, nonlinearly changed to get better dynamics of balance between global and local search capabilities, so a fuzzy particle swarm optimization (FPSO) algorithm is proposed in this paper, which combines particle's normalized current best performance evaluation (*NCBPE*) with the linearly decreased inertia weight in standard PSO to modify the inertia weight in FPSO with fuzzy rules [21, 22], so as to avoid the disadvantage that inertia weight is only affected by the number of iteration. Furthermore, as for the problem that particles are easy to being trapped in local optima, based on the FPSO, the crossover thought in genetic algorithm [23] is also introduced to enhance the diversity of particles, whose operation is to imitate genetic crossover between any two adjacent particles. On the basis of these facts, an improved fuzzy particle swarm optimization (IFPSO) algorithm is proposed. The sum of adjusting time with another number which represents the integral for sum of system error absolute value and the squared governor output is taken as the fitness function of the proposed algorithm.

And then, the proposed IFPSO, as well as some other algorithms including Ziegler Nichols (ZN) algorithm, fuzzy PID algorithm and standard PSO, has been examined and tested to optimal tuning of PID parameters for two different types of WTRS systems, which are the typical WTRS system and a kind of WTRS system with respect to the water flow inertia effect. The performances of these algorithms have been compared under frequency disturbances and load disturbances, and the simulation results reveal that, under all of the running conditions, IFPSO can find the solution with the highest quality which results in the hydroelectric generating set control system achieving the stable state with small overshoot and shorter settling time.

The rest of this paper is organized as follows. The mathematical models of two types of WTRS system are introduced in Section 2. Section 3 describes the proposed IFPSO algorithm with fuzzy and crossover thought and the proposed fitness function in details. The experiments and results for the WTRS systems are presented in Section 4 and the conclusions are given in Section 5.

2. MATHEMATICAL MODELS OF WATER TURBINE REGULATING SYSTEM

WTRS including control system and controlled objects is a non-minimum phase system with complex structure and strong nonlinearity, which is also an integrated control system combining flow, electrical and mechanical properties etc. The control system is mainly made up of speed governor and electro-hydraulic servo system, and the controlled objects consisted of turbine units and generator [1]. The structure of a typical WTRS is illustrated in Fig. (**1**).

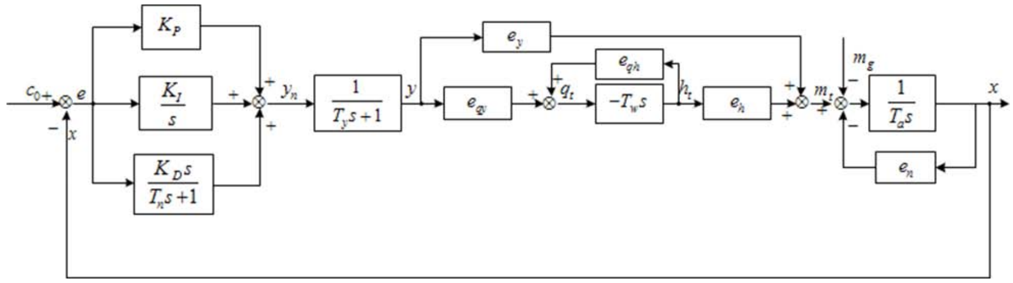


Fig. (1). Mathematical model of a typical WTRS system.

2.1. Model of Turbine Governor

As shown in Fig. (**1**), *c* and y_n represent the unit speed control instructions and the output signal of PID controller, respectively; *x* is the rotation speed feedback control signal; *e* is the error signal generated by unit speed input and output command signal; *y* is the guide vane opening signal. Hydraulic turbine governor is composed of electro-hydraulic servo system and the controller which adopts parallel PID regulation law.

The transfer function of PID controller is described as follows [1, 24].

$$G_{PID}(s) = \frac{y_n(s)}{e(s)} = K_P + \frac{K_I}{s} + \frac{K_D s}{T_n s + 1} \tag{1}$$

where K_P is the proportion adjustment coefficient, K_I is the integral adjustment coefficient (s^{-1}), K_D is the differential adjustment factor (s), *s* is the Laplace operator and T_n represents the differential time constant, usually, T_n=0.1*K_D.

As the actuator of water turbine, servomechanism can turn the controller's output into hydraulic signal. And then this signal is enhanced gradually through the guiding device, auxiliary servomotor and main reaction servomotor for producing enough power to drive the turbine guide vane, so as to adjust the speed and power of hydro-turbine generator unit sets (HGU). The transfer function of the electro-hydraulic servo system is expressed as [1, 24]:

$$G_s(s) = \frac{y(s)}{y_n(s)} = \frac{1}{T_y s + 1} \tag{2}$$

where T_y is the relay device related time constant (s).

2.2. Model of Penstock System

The water flow and the aqueduct wall of water diversion system are assumed to be rigid if the fluctuation is small, the transfer function is [1, 24]:

$$G_h(s) = \frac{h_t(s)}{q_t(s)} = -T_w s \tag{3}$$

where T_w is inertial time constant for water (s), $q_t(s)$ and $h_t(s)$ are the Laplace transform of water flow $q_t(t)$ and effective water head $h_t(t)$, respectively.

2.3. Model of Hydraulic Turbine System

WTRS system is a complicated system with serious nonlinearity, it's necessary to simplify it and ignore some secondary factors when we build the mathematical model. And it seems impossible that there exists a simulated experiment which can completely simulate the actual working process. But only when the system suffers small fluctuation of dynamic process (*i.e.* the disturbance and the parameter variation are small in the system), the system can be thought as a linear one.

In this case, the mathematical model of hydraulic turbine can be expressed as [1, 24]:

$$\begin{cases} m_t = e_x x + e_y y + e_h h_t \\ q_t = e_{qx} x + e_{qy} y + e_{qh} h_t \end{cases} \tag{4}$$

where e_y, e_h, e_x, e_{qy}, e_{qh}, and e_{qx} are the turbine transfer coefficients of controlled objects, m_t is the generated water torque.

2.4. Model of Generator System

In this part, the simplified first-order generator model is adopted and the transfer function is deduced as:

$$G_g(s) = \frac{x(s)}{m_t(s) - m_g(s)} = \frac{1}{T_a s + e_n} \tag{5}$$

$$e_n = e_g - e_x \tag{6}$$

where $m_g(s)$ is the Laplace transform of load disturbance torque $m_g(t)$, T_a is the generator inertial time constant (s), e_g is the generator load adjustment coefficient, e_n is the controlled system self-regulation coefficient.

When the guide vane opening changes quickly, *e.g.* it decreases rapidly, due to the effect of inertia of water, flow velocity increases with the decreasing of guide vane opening, which will make the turbine torque increase. In the consideration of this phenomenon, e_{qx}, the transmission coefficient of discharge to speed, is introduced into the typical model to form the WTRS mathematical model with respect to water flow inertia effect which is shown in Fig. (**2**).

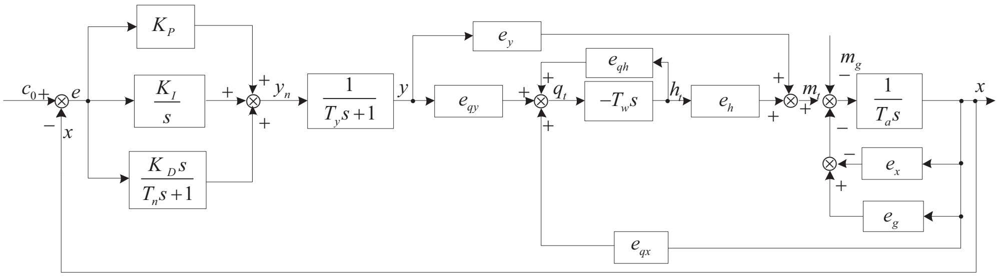


Fig. (2). Mathematical model of WTRS with respect to water flow inertia effect.

3. THE IMPROVED FUZZY PARTICLE SWARM OPTIMIZATION ALGORITHM

3.1. The Standard Particle Swarm Optimization Algorithm

In PSO algorithm, the population is initialized to a group of random particles and each particle has its own speed and position which are represented as $v_i^t = (v_{i1}^t, v_{i2}^t, \ldots, v_{id}^t)$ and $x_i^t = (x_{i1}^t, x_{i2}^t, \ldots, x_{id}^t)$, respectively. Thereinto, $i = 1, 2, ...,$

m, where *m* is the number of particles; $d = 1, 2, ..., D$, *D* is the dimensions of individual particle. The search process in a *D*-dimensional space for a swarm composed of *m* particles is formed. During this process, the quality of each particle is evaluated by the fitness function; the position of particle is affected primarily by two factors. One is p_{best}, the particle's personal best position that is represented as $p_i^t = (p_{i1}^t, p_{i2}^t, \ldots, p_{id}^t)$, and the other is g_{best}, global historical best position, which can be represented as $p_g^{\mathrm{t}} = (p_{g1}^{\mathrm{t}}, p_{g2}^{\mathrm{t}}, \ldots, p_{gd}^{\mathrm{t}})$. Particle's velocity and position updating formulas are shown below.

$$\begin{aligned} v_{id}^{t+1} &= w^t \cdot v_{id}^t + c_1 \cdot r_1^t \cdot (p_{id}^t - x_{id}^t) \\ &+ c_2 \cdot r_2^t \cdot (p_{gd}^t - x_{id}^t) \end{aligned} \tag{7}$$

$$x_{id}^{t+1} = x_{id}^t + v_{id}^{t+1} \tag{8}$$

$$w^t = w_{\max} - t(w_{\max} - w_{\min}) / t_{\max} \tag{9}$$

where *t* denotes the current iteration; w^t is the inertia weight, normally, $w_{\max} = 0.9$, $w_{\min} = 0.4$; $t_{\max}$ is the number of maximum generation; c_1 and c_2 are two constants, which are the learning weights following individual optimal and global optimal, respectively; r_1^t and r_2^t are two random numbers drawn from the interval (0, 1).

3.2. Fuzzy Particle Swarm Optimization (FPSO) Algorithm

In the standard PSO, the inertia weight decreases linearly during the process of search. However, by this linear decreasing inertia weight strategy, it is hard, if not impossible, to accurately reflect the optimization process for the following two deficiencies.

i. At the beginning of iterations, a larger value of weight coefficient contributes to strengthen global exploring ability of particles, but while the particles are very close to the global optimal, the larger speed may lead them to deviating from the global optimal and straying from the correct direction, thus reducing the search accuracy.
ii. In the late iterations, all particles being gathered nearby the optimal value, it's a smaller inertial factor that makes particles to do fine search around the optimal scope. But at this moment, the smaller speed caused by the smaller inertia weight may make solutions trend to be identity and trap in local optimal.

Therefore, it's clear that there exists the deficiency about the choice of the inertia weight based on the number of iterations in standard PSO. Therefore, in order to improve the performance, the inertia weight should be dynamically, nonlinearly changed to get better dynamics of balance between global and local search capabilities [21]. In this paper, the fuzzy thought is introduced into PSO algorithm to solve the problem of selecting inertia weight in different stages of searching process.

In FPSO algorithm, the inertia weight is influenced by two factors, one is the current linearly decreasing inertia weight, and the other is the normalized current best performance evaluation (*NCBPE*) [21]. In the minimization optimization problem, *NCBPE* can be calculated as Eq. (10).

$$NCBPE = \frac{CBPE - CBPE_{\min}}{CBPE_{\max} - CBPE_{\min}} \tag{10}$$

where *CBPE* represents the current best performance evaluation of particles, $CBPE_{\max}$ is the non-optimal *CBPE* at the beginning of interations, $CBPE_{\min}$ is the estimated minimum at the end of interations.

Combining *NCBPE* and linearly decreasing inertia weight w^t to modify the weight coefficient w_c^t in FPSO. w_c^t can nonlinearly, dynamically change to keep a tradeoff between the particle global exploring ability and local search ability. That is to say, *NCBPE* and current linearly decreasing inertia weight w^t are selected as the inputs of the fuzzy control and the revised inertia weight w_c^t is the output. In the Table **2**, the inputs and output are defined to have the same fuzzy sets: small (S), medium (M) and large (L). The membership functions and the critical parameters of three variables are list in the Table **1**.

Table 1. The border of membership functions.

	zmf	trimf	smf
NCBPE	(0,0.06)	(0.05,0.45/2,0.4)	(0.3,1)
w^t	(0.4,0.6)	(0.5,1.3/2,0.8)	(0.7,0.9)
w_c^t	(0,0.3)	(0.2,0.5,0.8)	(0.7,1)

Several rules need to be considered in the process of rule's setting, which are as followings and as Table **2**:

i. During the initial stage of iterations, at the moment that the value of w^t is larger, if *NCBPE* is also large, particles are deemed to be far away from the optimal value, and then particles need to speed up and do a comprehensive exploration in the global scope; on the contrary, if *NCBPE* is medium at this moment, it means that the distance between the particles and optimal value is medium, w_c^t should be set to an appropriate value; and if *NCBPE* is small, it represents the particles are close to the optimal value and w_c^t should be given a smaller value to avoid the speed too fast at the same time.

ii. During the middle stage of iterations, the value of w^t is medium, and if *NCBPE* is large or medium at this point, w_c^t should be taken to a medium value; contrarily, if *NCBPE* is small, it means the particles are very close to the optimal value and w_c^t should be set to a small value, which contributes to increase the possible of fine searching near the optimal solutions.

iii. During the late stage of iterations, w^t is given a small value. At this moment, if *NCBPE* is large or medium, w_c^t should be taken to a large value to speed up for the sake of finding the optimal value; rather, if *NCBPE* value is small, w_c^t should be set to a medium value.

Table 2. The fuzzy control rules.

NCBPE w_c^t w^t	**S**	**M**	**L**
S	M	L	L
M	S	M	M
L	S	M	L

3.3. Improved Fuzzy Particle Swarm Optimization (IFPSO) Algorithm

PSO can fast convergence in the early iterations, however convergence speed gradually slows down along with iteration process and the loss of diversity leads particles to tend to be identity; hence, they are easy to fall into local optimal at late iterations. In order to maintain the diversity of population and enhance the global search ability of particles, in this article, the crossover thought in genetic algorithm (GA) is introduced into FPSO, increasing the diversity of swarm to let it step out the local optimal through the genetic exchange between adjacent particles.

Combining the fuzzy thought and crossover thought, an improved fuzzy particle swarm optimization (IFPSO) algorithm is proposed, whose update formulas of particle's velocity and position are as below.

$$\begin{aligned} v_{id}^{t+1} = w_c^t \cdot v_{id}^t + c_1 \cdot r_1^t \cdot (p_{id}^t - x_{id}^t) \\ + c_2 \cdot r_2^t \cdot (p_{gd}^t - x_{id}^t) \end{aligned} \tag{11}$$

$$x_{id}^{t+1} = x_{id}^t + v_{id}^{t+1} \tag{12}$$

$$\begin{cases} \left.\begin{aligned} x_{id}^{t+1} &= (1-alfa)\cdot x_{id}^{t+1} \\ &\quad + alfa\cdot x_{(i+1)d}^{t+1}, \\ x_{(i+1)d}^{t+1} &= (1-alfa)\cdot x_{(i+1)d}^{t+1} \\ &\quad + alfa\cdot x_{id}^{t+1} \end{aligned}\right\} & if\ rand(n) < P_c \\ x_{id}^{t+1} = x_{id}^{t+1} & else \end{cases} \tag{13}$$

where,

$$\begin{cases} v_{id}^{t+1} = v_{\max} & if \quad v_{id}^{t+1} > v_{\max} \\ v_{id}^{t+1} = -v_{\max} & if \quad v_{id}^{t+1} < -v_{\max} \end{cases}$$

$$\begin{cases} x_{id}^{t+1} = x_{\max} & if \quad x_{id}^{t+1} > x_{\max} \\ x_{id}^{t+1} = x_{\min} & if \quad x_{id}^{t+1} < x_{\min} \end{cases}$$

where *alfa* and *rand*(*n*) are two random numbers within the interval (0, 1). P_c is the crossover probability, which will affect the convergence efficiency. A larger one will increase the possibility that particles generate new individuals, which renders convergence process more efficient. However, if the crossover probability is too large, the optimal value that has been found may be lost, generally P_c =0.8. $v_{\max}$ is the boundary value of speed. $x_{\max}$ and $x_{\min}$ are upper limit and lower limit, respectively.

By the means of above two strategies added in the standard PSO, the proposed IFPSO can be summarized as the following.

1. Randomly initialize the particles position and velocity and set the various parameters of population.
2. Calculate the fitness, and update p_{best} and g_{best}.
3. Evaluate the value of *NCBPE* according to Eq. (10), and set the fuzzy control parameters and fuzzy rules.
4. Update the particle velocity and position according to Eqs. (11) and (12).
5. Judge whether the velocity and position are beyond the boundary. The velocity and position are limited within the range, if they are against the boundary.
6. Calculate particle fitness again, if the current particle fitness is better than p_{best}, then replace p_{best} with current fitness; If the current global optimal value is superior to global optimal, then replace g_{best} with the current global optimal.
7. If iterations have reached to the maximum, stop searching and output the global optimal, otherwise, return to step 3.

3.4. IFPSO Algorithm Fitness Function

Fitness is the particle quality evaluation standard, with which we can seek to find the optimized PID values. The optimized controller should make the system obtaining good performance, such as faster response speed, smaller overshoot and lower settling time, in the process of dynamic response and make the steady-state error converging to zero finally. The sum of adjusting time with another number which is the integral of sum of system error absolute value and the squared governor output is taken as the fitness function of the proposed algorithm. Moreover, once the system appears overshoot, the overshoot part will be taken as a part of fitness function.

$$J = \int_0^{T_s} (p_1\,|\,e(t)\,|+p_2 y_n^2(t))dt + p_3\cdot t_r \tag{14}$$

$$\begin{aligned} & if\ \ e(t) < 0 \\ & J = \int_0^{T_s} (p_1 \,|\, e(t) \,|\, + p_2 y_n^2(t) + p_4 \,|\, e(t) \,|\,) dt + p_3 \cdot t_r \end{aligned} \quad \textbf{(15)}$$

where $y_n\ (t)$ is the output of PID controller; t_r is regulating time; p_1, p_2, p_3 and p_4 are the weight coefficients, and p_4 is much larger than p_1; T_s represents the simulation time.

4. THE SIMULATION EXPERIMENTS AND RESULTS

Regarding the hydraulic turbine regulating system, the main goals of governor parameters setting are to ensure unit frequency can fast track the given frequency value accurately and eliminate the effects on the frequency of unit caused by load disturbances. Therefore, in the process of parameters optimization, it's two kinds of transition processes that need to be considered. One is the transition process caused by the excitation of given frequency, generally, which is produced by a frequency step signal in the simulation test. And the other is caused by load disturbance which is generated by a load disturbance step signal.

4.1. Experimental Data

In this paper, the two WTRS systems, as illustrated in Figs. (**1** and **2**), are tested in simulation experiments, which are set as case 1 and case 2, respectively. And the required data of WTRS systems are listed in Table **3**. The parameters setting of the proposed algorithm is shown in Table **4**. In Table **5**, the values of $CBPE_{\max}$ and $CBPE_{\min}$ for two cases are listed.

Table 3. Parameters of two different types of WTRS systems.

Case No.	Parameters of water turbine regulating system										
	T_y(s)	T_a(s)	T_w(s)	e_y	e_h	e_{qx}	e_{qy}	e_{qh}	e_g	e_x	e_n
1	0.3	5.72	0.83	1.40	0.35	-	1.23	0.13	-	-	0.45
2	0.3	5.72	0.83	0.7212	1.5505	-0.0322	0.8730	0.5708	0	-0.8771	-

Table 4. Parameters setting for IFPSO.

Parameters	Values	Parameters	Values
m	30	$w_{\min}$	0.4
$t_{\max}$	100	P_c	0.8
c_1	2	p_1	0.999
c_2	2	p_2	0.001
$w_{\max}$	0.9	p_3	2
$x_{\max}$	10	p_4	100
$x_{\min}$	0		

Table 5. Values of $CBPE_{\max}$ and $CBPE_{\min}$ for two cases.

	Case 1		Case 2	
	$CBPE_{\max}$	$CBPE_{\min}$	$CBPE_{\max}$	$CBPE_{\min}$
5% frequency-disturbance	18.0	6.0	67.0	20.0
10% frequency-disturbance	26.0	10.0	100.0	35.0
5% load-disturbance	6.0	4.0	45.0	20.0
10% load-disturbance	10.0	5.0	60.0	30.0

4.2. Comparison of Optimization Methods Under No-load Frequency Disturbance Condition

In order to verify the performance of the proposed algorithm when it is applied to optimize the turbine governor parameters, the simulation experiments for hydro-generator units under 5% and 10% frequency disturbance are carried out. The step disturbances of given speed with corresponding amplitudes are employed to excite the systems. The average best fitness of IFPSO and PSO are compared in Figs. (**3**, **5**, **7**, **9**), both of which are repeated for 50 trials for overcoming the randomness of the heuristic algorithms. In addition, the traces of turbine speed deviation obtained by

IFPSO, standard PSO, ZN algorithm and fuzzy PID algorithm are compared in Figs. (**4**, **6**, **8**, **10**), respectively. Some results considering best fitness (*Best F*), the standard deviation of *Best F* (*STDEV*), overshoot (*Overshoot*) and settling time (t_s) are listed in Tables **6** and **7**.

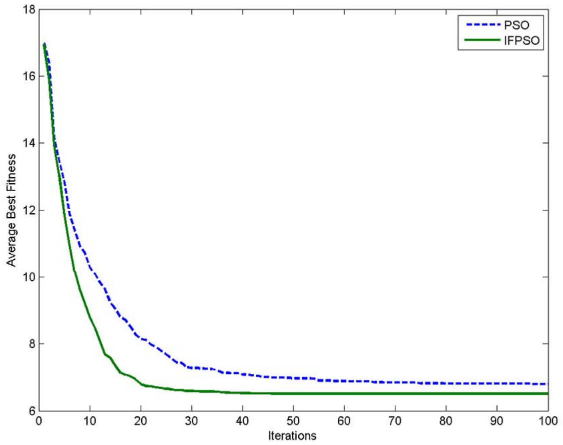


Fig. (3). Average best fitness under 5% frequency disturbance (case 1).

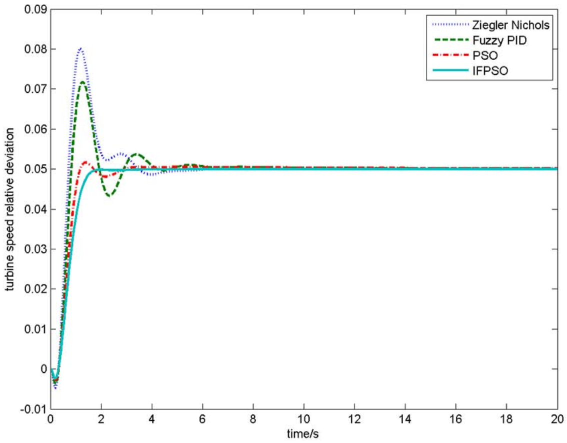


Fig. (4). Speed deviation under 5% frequency disturbance (case 1).

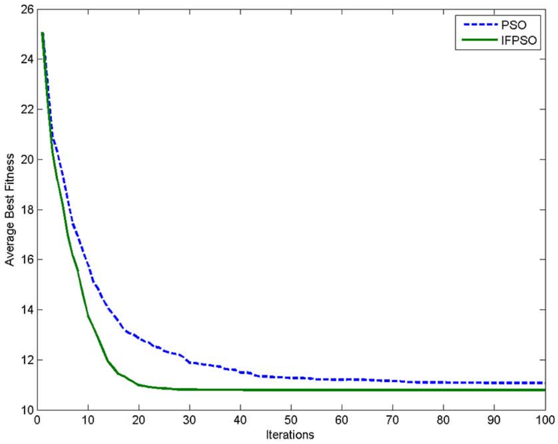


Fig. (5). Average best fitness under 10% frequency disturbance (case 1).

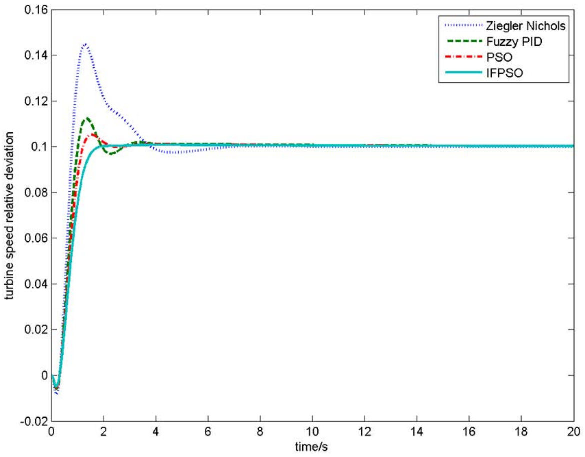


Fig. (6). Speed deviation under 10% frequency disturbance (case 1).

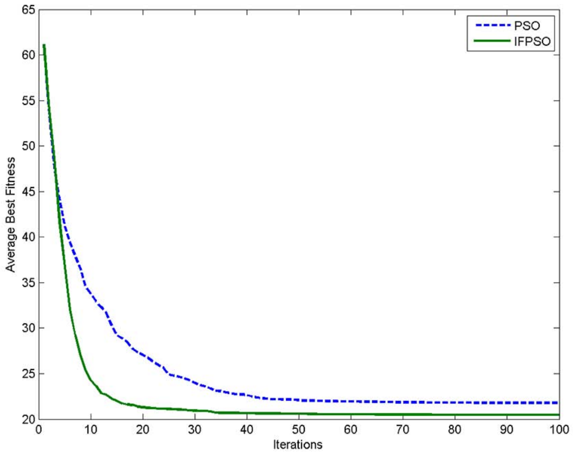


Fig. (7). Average best fitness under 5% frequency disturbance (case 2).

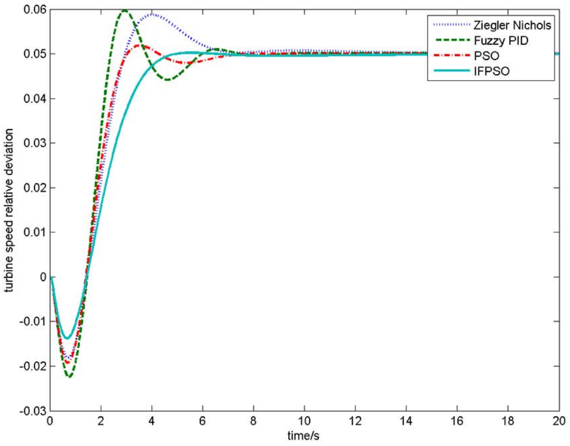


Fig. (8). Speed deviation under 5% frequency disturbance (case 2).

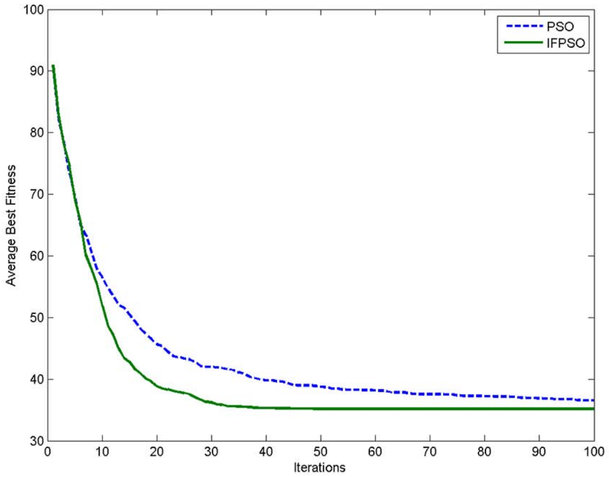


Fig. (9). Average best fitness under 10% frequency disturbance (case 2).

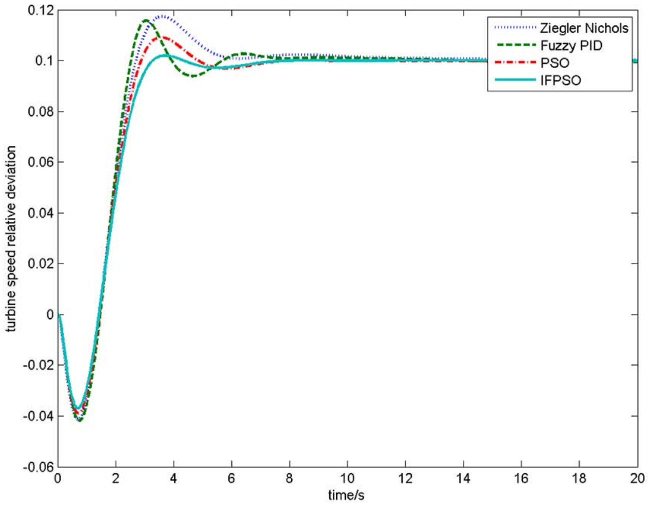


Fig. (10). Speed deviation under 10% frequency disturbance (case 2).

Table 6. Simulation results of case 1 under frequency disturbance.

Conditions	5% frequency disturbance				10% frequency disturbance			
Algorithms	**ZN**	**Fuzzy PID**	**PSO**	**IFPSO**	**ZN**	**Fuzzy PID**	**PSO**	**IFPSO**
Best F	-	-	6.7981	6.5089	-	-	11.0601	10.7856
STDEV	-	-	0.9613	0.0048	-	-	0.3215	0.0031
Overshoot	0.5824	0.4314	0.0302	0	0.4461	0.1206	0.0500	0.0061
t_s(s)	5.6300	6.1400	4.0700	3.7200	6.5800	4.1300	4.0400	2.0300
K_P	8.7993	-	5.6364	5.1669	7.5444	-	5.8973	5.0805
K_D(s)	2.6006	-	1.0825	1.1979	2.3503	-	1.3968	1.1833
K_I(s^{-1})	7.4432	-	0.5439	0.3901	6.0544	-	0.5555	0.4568

From the profile of average best fitness, it is obvious that,compared with standard PSO, IFPSO acquires better convergence results based on the proposed fitness function with faster convergence speed. The final average best fitness obtained by IFPSO is smaller than that obtained by PSO, which means that the particles are easier to jump out from local optimal to find the better global optimal within the global range at the late iterations of the search process by using IFPSO.

Table **6** shows that, for the typical WTRS system model (case 1), whether under 5% or 10% frequency disturbance conditions, the system dynamic performance obtained by IFPSO algorithm outperforms than those by other three algorithms, which presents a shorter settling time and smaller overshoot, although, sometimes the system has a slower response speed when under IFPSO based tuning. Table **7** shows the compared results of the aforementioned four algorithms for the WTRS with respect to water flow inertia effect (case 2). It is found that IFPSO-based strategy obtains the best performance in terms of settling time and overshoot.

Table 7. Simulation results of case 2 under frequency disturbance.

Conditions	5% frequency disturbance				10% frequency disturbance			
Algorithms	ZN	Fuzzy PID	PSO	IFPSO	ZN	Fuzzy PID	PSO	IFPSO
Best F	-	-	21.7863	20.4705	-	-	36.5761	35.1762
STDEV	-	-	3.1841	5.9931	-	-	2.4346	0.0016
Overshoot	0.1724	0.1927	0.0369	0.0066	0.1708	0.1646	0.0919	0.0195
t_s(s)	12.0100	8.9800	7.7100	6.6400	12.2400	12.1900	8.8300	8.0800
K_P	3.7707	-	3.5182	3.0671	3.8967	-	3.6798	3.4562
K_D(s)	1.4497	-	1.8396	1.1373	1.7818	-	1.7644	1.7436
K_I(s^{-1})	0.6129	-	0.5002	0.4271	0.6392	-	0.5202	0.4933

From the above it is observed that, when compared with PSO algorithm, ZN algorithm and fuzzy PID algorithm, the proposed IFPSO can obtain better overshoot and settling time, thereby justifying the rationality and superiority of the proposed algorithm for optimization tuning of WTRS systems under frequency disturbance conditions.

4.3. Comparison of Optimization Methods Under Load Disturbance Condition

In order to assess the validity and robustness of the proposed algorithm, load disturbance condition is also considered. Different step disturbances of load are adopted to excite systems. The experiments for systems under 5% and 10% load disturbance conditions are performed within 20 s and repeated 50 times.

From the compared results for fitness function convergence situations of IFPSO and PSO shown in Figs. (**11**, **13**, **15**, **17**), we can find that, the final average best fitness value obtained by IFPSO is smaller than that by PSO. It illustrates that IFPSO possesses superior ability in obtaining higher-quality optimal value with faster convergence speed compared with PSO algorithm in the searching process and it is capable of stepping out the local optimal.

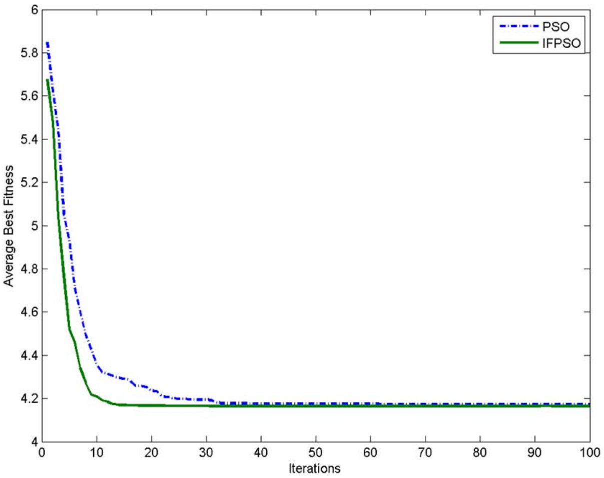


Fig. (11). Average best fitness under 5% load disturbance (case 1).

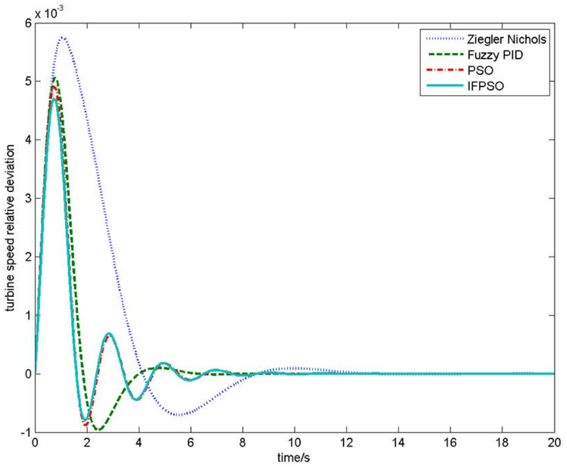


Fig. (12). Speed deviation under 5% load disturbance (case 1).

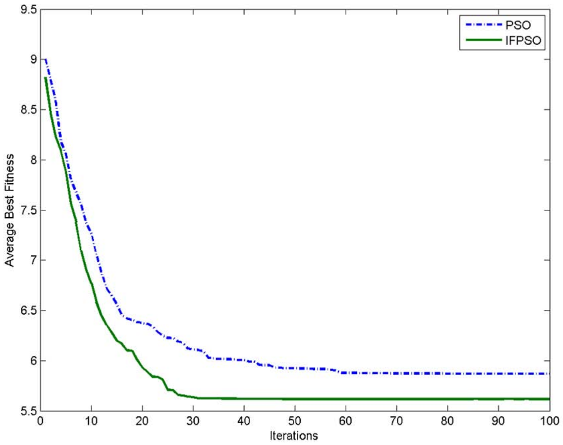


Fig. (13). Average best fitness under 10% load disturbance (case 1).

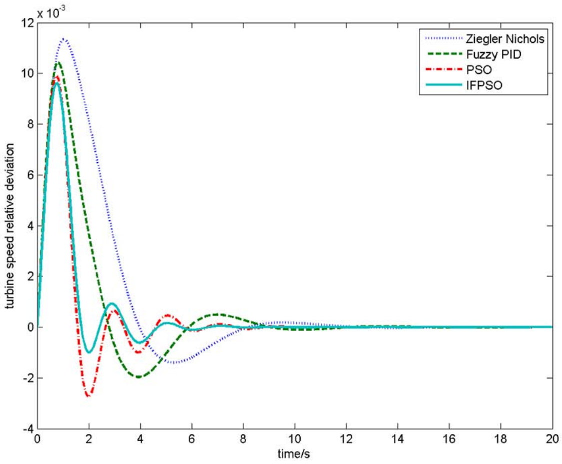


Fig. (14). Speed deviation under 10% load disturbance (case 1).

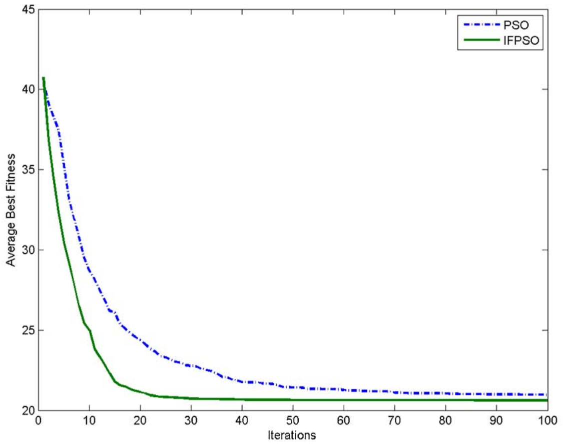


Fig. (15). Average best fitness under 5% load disturbance (case 2).

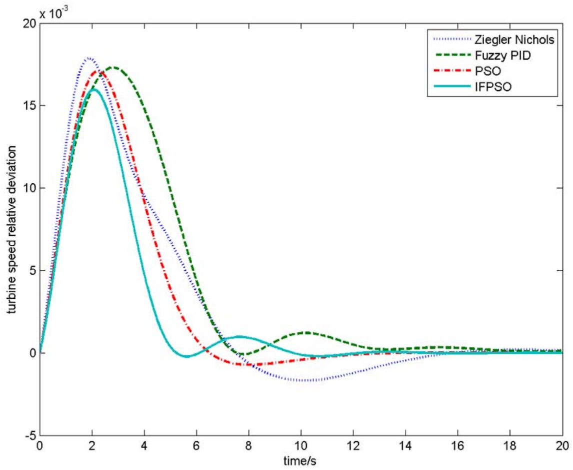


Fig. (16). Speed deviation under 5% load disturbance (case 2).

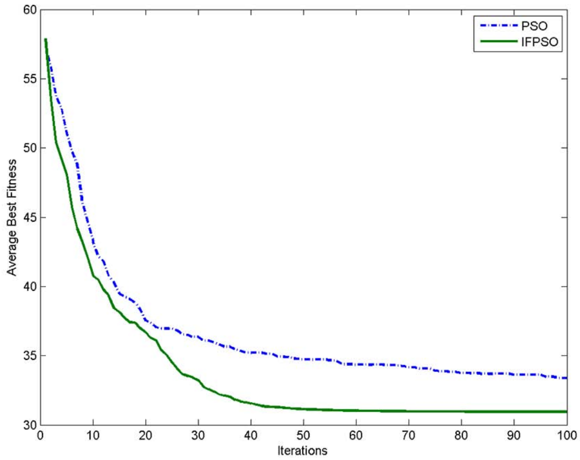


Fig. (17). Average best fitness under 10% load disturbance (case 2).

Figs. (**12**, **14**, **16**, **18**) show the speed deviation transient process of the two WTRS systems by using the optimal solutions of IFPSO algorithm and other three optimization approaches. Obviously, from these figures, it can be found that with the regulation of IFPSO based controller, the fluctuation caused by the load disturbance is fast weakened and gradually disappear, and then achieve the steady state as desired soon.

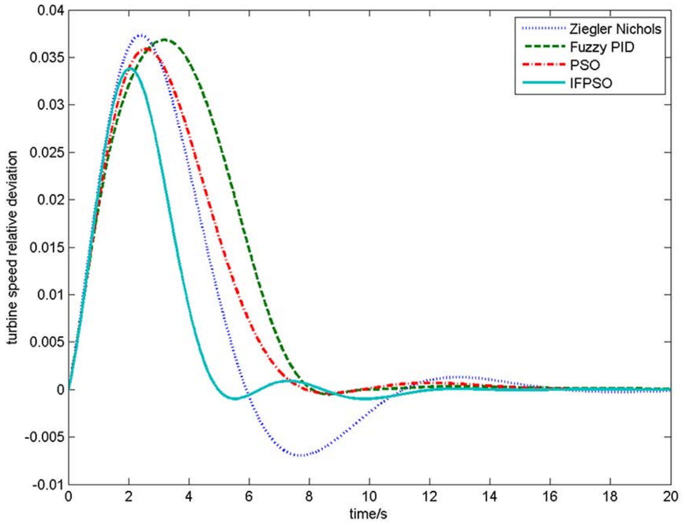


Fig. (18). Speed deviation under 10% load disturbance (case2).

Some experimental simulation results under different load conditions are listed in Tables **8** and **9**. It is seen that compared with the other three approaches, the proposed IFPSO gets a great improvement in terms of settling time with smaller overshoot.

Table 8. Simulation results of case 1 under load disturbance.

Conditions	5% load disturbance				10% load disturbance			
Algorithms	**ZN**	**Fuzzy PID**	**PSO**	**IFPSO**	**ZN**	**Fuzzy PID**	**PSO**	**IFPSO**
Best F	-	-	4.1738	4.1636	-	-	5.8695	5.6146
STDEV	-	-	0.0009	0.0207	-	-	0.8329	0.0126
Overshoot	0.0057	0.0051	0.0049	0.0047	0.0113	0.0104	0.0099	0.0096
t_s(s)	12.5400	6.0700	8.4800	8.4300	12.2100	11.9700	8.7300	7.6400
K_P	4.8256	-	9.8860	9.9833	5.0064	-	8.7795	9.4568
K_D(s)	1.8765	-	2.3138	2.3321	1.8885	-	2.8263	2.2297
K_I(s^{-1})	3.1024	-	8.3454	8.2046	3.3180	-	10.0000	7.5664

Table 9. Simulation results of case 2 under load disturbance.

Conditions	5% load disturbance				10% load disturbance			
Algorithms	**ZN**	**Fuzzy PID**	**PSO**	**IFPSO**	**ZN**	**Fuzzy PID**	**PSO**	**IFPSO**
Best F	-	-	20.9655	20.6289	-	-	33.3868	30.9531
STDEV	-	-	0.4532	0.0003	-	-	1.5011	1.4187
Overshoot	0.0177	0.0173	0.0171	0.0159	0.0373	0.0337	0.0358	0.0337
t_s(s)	17.2500	17.2500	12.1100	11.6600	16.7300	13.9500	15.5700	11.8600
K_P	3.8440	-	4.0978	4.7395	3.8503	-	3.6976	4.6691
K_D(s)	2.8429	-	1.7578	2.0493	1.3824	-	1.0000	2.2135
K_I(s^{-1})	1.2994	-	1.2566	1.3759	1.3405	-	0.9642	1.4748

Besides, under all of the running conditions, the overshoot level obtained by IFPSO based controller is lower than those obtained by other three algorithms based controllers. Obviously, the lower overshoot results in the better performance of WTRS, therefore the IFPSO based controller is confirmed to be more suitable for the tuning of WTRS system.

CONCLUSION

In this paper, the challenge of parameter optimization of WTRS system is studied. An improved fuzzy particle swarm optimization (IFPSO) algorithm is proposed by incorporating the fuzzy control thought and the crossover thought in genetic algorithm (GA) based on the standard PSO. The simulation results indicate that the proposed IFPSO outperforms standard PSO on convergence speed and precision based on the proposed fitness function. And the IFPSO algorithm has been proven to be more effective for solving parameter optimization problem by experiments for two different types of WTRS systems when compared with standard PSO, Ziegler Nichols algorithm and fuzzy PID algorithm. Furthermore, the experiment results reveal that, both under frequency and load disturbance conditions, the proposed IFPSO could lead to the best quality solutions. And under the optimized solutions obtained by IFPSO, the dynamic transition process of system is stable with smaller overshoot and shorter settling time.

CONFLICT OF INTEREST

The authors confirm that this article content has no conflict of interest.

ACKNOWLEDGEMENTS

The authors would like to thank the editors and the reviewers for their constructive comments. This work was supported by the National Natural Science Foundation of China (No. 51507024 and 61263030) and Science and Technology Research Project of Chongqing Municipal Education Commission (No. KJ1500401).

REFERENCES

[1] S. Wei, *Simulation of hydraulic turbine regulation system.,* Huazhong University of Science and Technology Press: Wuhan, 2011.

[2] L. Wang, L. Yang, and Y. Tian, "Study on energy efficient operation of air-conditioning cooling water system", *Open Electr. Electron. Eng. J.,* vol. 9, pp. 53-60, 2015.
[http://dx.doi.org/10.2174/1874129001509010053]

[3] S. Li, and Y. Peng, "Field analysis on magnetic transmission mechanism of downhole turbine generator", *Open Electr. Electron. Eng. J.,* vol. 9, pp. 43-52, 2015.
[http://dx.doi.org/10.2174/1874129001509010043]

[4] T. Wu, J. Yang, and Q. Miao, "DC side ripple voltage analysis and fuzzy PI control strategy research for APF", *Open Electr. Electron. Eng. J.,* vol. 9, pp. 263-272, 2015.
[http://dx.doi.org/10.2174/1874129001509010263]

[5] Z. Shen, "Optimum setting of hydroturbine governor pole assignment method", *East China Tech. Univ. Water Res.,* vol. 4, pp. 8-18, 1984.

[6] Z. Li, and O. Malik, "An orthogonal test approach based control parameter optimization and its application to a hydro-turbine governor", *IEEE Power Energy Society,* vol. 12, no. 4, pp. 388-393, 1997.

[7] S. Jiang, Q. Chen, and W. Cai, "Simulation optimization strategy for parameters of hydro-governor", *Proc. CSEE,* vol. 28, pp. 102-106, 2008.

[8] M.K. Singh, R. Naresh, and D.K. Gupta, "Optimal tuning of temporary droop governor of hydro power plant using genetic algorithm", *Int. Conf. Energy Efficient Technol. Sustain,* pp. 1132-1137, 2013.
[http://dx.doi.org/10.1109/ICEETS.2013.6533545]

[9] Y. Luo, and X. Che, "Tuning PID control parameters on hydraulic servo control system based on differential evolution algorithm", *Int. Conf. Adv. Comput. Control,* vol. 58, pp. 348-351, 2010.
[http://dx.doi.org/10.1109/ICACC.2010.5486842]

[10] C. Li, J. Zhou, and J. Xiao, "PID parameter optimization of excitation control systems by using improved gravitational search algorithm", *J. Huazhong Univ. Sci. Tech.,* vol. 40, no. 10, pp. 119-122, 2012. [Natural Science Edition].

[11] L. Zhang, H. Bian, and G. Chen, "Perpendicularity errors evaluation of spatial lines based on PSO algorithm using an improved joint normal transform method for modeling stochastic dependence in power system", *Open Electr. Electron. Eng. J.,* vol. 9, pp. 380-385, 2015.
[http://dx.doi.org/10.2174/1874129001509010380]

[12] Y. He, S. Yang, and Q. Xu, "Short-term cascaded hydroelectric system scheduling based on chaotic particle swarm optimization using improved logistic map", *Commun. Nonlinear Sci. Numer. Simul.,* vol. 18, pp. 1746-1756, 2013.
[http://dx.doi.org/10.1016/j.cnsns.2012.11.003]

[13] H. Li, Z. Li, and R. He, "Rock burst risk evaluation based on particle swarm optimization and BP neural network", *J. Min. Saf. Eng,* vol. 31, no. 2, pp. 203-207, 2014.

[14] X. Xia, J. Liu, and Z. Hu, "An improved particle swarm optimizer based on tabu detecting and local learning strategy in a shrunk search space", *Appl. Soft Comput.,* vol. 23, pp. 76-90, 2014.
[http://dx.doi.org/10.1016/j.asoc.2014.06.012]

[15] J. Song, H. Zhao, and Y. Jiang, "Application of particle swarm optimization in the optimal water allocation of Miyun Reservoir", *South-to-*

North Water Trans. Water Sci. Technol, vol. 13, no. 2, pp. 378-381, 2015.

[16] H. Shayeghi, M. Mahdavi, and A. Bagheri, "An improved DPSO with mutation based on similarity algorithm for optimization of transmission lines loading", *Energy Convers. Manage.,* vol. 51, pp. 2715-2723, 2010. [http://dx.doi.org/10.1016/j.enconman.2010.06.007]

[17] H. Fang, and Z. Shen, "Optimal hydraulic turbogenerators PID governor tuning with an improved particle swarm optimization algorithm", *Proc. CSEE,* vol. 25, pp. 102-106, 2005.

[18] X. Xia, J. Liu, and Z. Hu, "An improved particle swarm optimizer based on tabu detecting and local learning strategy in a shrunk search space", *Appl. Soft Comput.,* vol. 23, pp. 76-90, 2014. [http://dx.doi.org/10.1016/j.asoc.2014.06.012]

[19] X. Wei, Q. Zhang, and H. Huang, "Optimization of hydro turbine governor parameters based on improved particle swarm", *Electric Power Sci. Eng,* no. 3, pp. 48-51, 2005.

[20] G. Chen, L. Liu, and P. Song, "Chaotic improved PSO-based multi-objective optimization for minimization of power losses and L index in power systems", *Energy Convers. Manage.,* vol. 86, pp. 548-560, 2014. [http://dx.doi.org/10.1016/j.enconman.2014.06.003]

[21] Y. Shi, and R.C. Eberhart, "Fuzzy adaptive particle swarm optimization", *Congr. Evol. Comput,* vol. 1, pp. 101-106, 2001.

[22] A.M. Abdelbar, S. Abdelshahid, and D.C. Wunsch Ii, "Fuzzy PSO: A Generalization of Particle Swarm Optimization", In: *Proc. Int. Joint Conf. Neural Netw,* Montreal, Canada, 2005, pp. 1086-1091. [http://dx.doi.org/10.1109/IJCNN.2005.1556004]

[23] G. Jahedi, and M.M. Ardehali, "Genetic algorithm-based fuzzy-PID control methodologies for enhancement of energy efficiency of a dynamic energy system", *Energy Convers. Manage,* vol. 52, pp. 725-732, 2011. [http://dx.doi.org/10.1016/j.enconman.2010.07.051]

[24] H. Fang, L. Chen, and Z. Shen, "Application of an improved PSO algorithm to optimal tuning of PID gains for water turbine governor", *Energy Convers. Manage.,* vol. 52, pp. 1763-1770, 2011. [http://dx.doi.org/10.1016/j.enconman.2010.11.005]