Engineering Applications of Artificial Intelligence ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

Particle swarm optimization with quantum infusion for system identification

Bipul Luitel, Ganesh K. Venayagamoorthy*

Real-Time Power and Intelligent Systems Laboratory, Missouri University of Science & Technology, Rolla, MO 65409, USA

ARTICLE INFO

Article history:
Received 2 February 2009
Received in revised form
26 November 2009
Accepted 16 January 2010

Keywords:
Adaptive IIR filter
DEPSO
Dynamical system
Power system
PSO
PSO–QI
Quantum principle
System identification

ABSTRACT

System identification is a challenging and complex optimization problem due to nonlinearity of the systems and even more in a dynamic environment. Adaptive infinite impulse response (IIR) systems are preferably used in modeling real world systems because of their reduced number of coefficients and better performance over the finite impulse response filters. Particle swarm optimization (PSO) and its other variants has been a subject of research for the past few decades for solving complex optimization problems. In this paper, PSO with quantum infusion (PSO–QI) is used in identification of benchmark IIR systems and a real world problem in power systems. PSO–QI's performance is compared with PSO and differential evolution PSO (DEPSO) algorithms. The results show that PSO–QI has better performance over these algorithms in identifying dynamical systems.

1. Introduction

Traditionally, least mean square (LMS) and other algorithms have been studied for the identification of linear and static systems (Widrow et al., 1976). But, almost all physical systems are nonlinear to certain extent and recursive in nature and hence it is more convincing to model such systems by using nonlinear models (Panda et al., 2007; Krusienski and Jenkins, 2005). Thus, nonlinear system identification has attracted attention in the field of science and engineering. Hence these are better modeled as infinite impulse response (IIR) models as they can provide better performance than a finite impulse response (FIR) filter with the same number of coefficients (Shynk, 1989a). Thus the problem of nonlinear system identification can also be viewed as a problem of adaptive IIR filtering. Also, IIR models are more efficient than the FIR models for implementation as they require less parameter and hence fewer computations for the same level of performance. However, there are few problems associated with the use of IIR models in identification of a system, such as instability of the algorithms, slow convergence and convergence to the local minimum (Netto et al., 1995). Different learning algorithms have been used in the past for nonlinear system identification. These techniques include use of neural network (Hongwei and Yanchun, 2005) and gradient based search techniques such as least mean square algorithm (Shynk, 1989(a)). Unfortunately, the error surface of such recursive systems such as a multi-machine power system (Kundur, 1993) tends to be multi-modal and hence traditional techniques of parameter approximation fail as they get trapped into local minimum and cannot attain the global minimum (Krusienski and Jenkins, 2005). Various algorithms that are implemented in the adaptive IIR filtering for system identification are described in (Netto et al., 1995).

Population based search algorithm such as genetic algorithm (GA) has also been used for the system identification. It uses a population of potential solutions encoded as chromosomes which go through genetic operations such as crossover and mutation to find the best solution (Kristinsson and Dumont, 1992). But its effectiveness is affected by the convergence time (the time it takes to find the global minimum). So to eliminate such deficiencies, population based stochastic optimization techniques have been discussed in various literatures. Particle swarm optimization (PSO) is one of the most known techniques (Kennedy and Eberhart, 1995). Application of PSO in the system identification has been discussed in Panda et al. (2007). In Lee et al. (2006), a method for the identification of nonlinear system and parameter optimization of the obtained input–output model has been described. The proposed method uses least squares support vector machines regression based on PSO. In another work, PSO has been used for optimizing the parameters of Elman neural network which is used for speed identification of ultrasonic motors (Hongwei and Yanchun, 2005). A modified form of PSO called as the self-organizing particle swarm optimization and its application in the system identification has been discussed in Shen and Zeng (2007). Radial basis function

* Corresponding author.
E-mail address: ganeshv@mst.edu (G.K. Venayagamoorthy).

0952-1976/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.engappai.2010.01.022

neural network (RBFNN) has been used for system identification in Chen et al. (2007), where a hybrid gradient-based PSO algorithm has been used to adjust the parameters of the RBFNN. In Liu et al. (2006), particle swarm optimization and quantum-behaved particle swarm optimization have been used for the system identification. Use of different types of stochastic optimization techniques in adaptive IIR filters and nonlinear systems has been explained in Krusienski and Jenkins (2005). Use of differential evolution (DE) and ant colony optimization (ACO) in IIR filter design has been presented in Karaboga (2005) and Karaboga et al. (2004), respectively. They also talk about the possible use of these approaches in system identification and other applications. But these algorithms have the tendency to get stuck in the local minimum when the complexity of the problem increases and in dynamic systems where time allowed for convergence is constrained. Hybrid algorithms are used to improve the performance by combining the best feature of participating individual algorithms. Differential evolution PSO (DEPSO) for digital filter design is discussed in Luitel and Venayagamoorthy (2008a).

However, identification of systems without prior structural information is still a challenge and new algorithms and approaches are being studied. Also, identification of nonlinear time varying systems is computationally intensive and many traditional techniques fail. In this paper, PSO with quantum infusion (PSO–QI) has been proposed to identify the pole zero parameters of an IIR system and in the identification of generator dynamics in a power system without prior structural information. PSO–QI has better performance and is robust in the fact that its convergence characteristics is less affected by the dimension of the problem and has more consistent convergence than other algorithms. Also, PSO–QI converges to a much lower value than PSO or DEPSO. In the identification of generator dynamics, PSO–QI performs the best, whereas PSO and DEPSO cannot approximate the system transfer function every time as is seen from the standard deviation over a number of trials. The major contributions of the paper are listed below:

- Identification of benchmark IIR systems with full and reduced order models using PSO–QI, which results in lower mean squared error and more consistent convergence.
- Application of PSO–QI in the identification of four generators in a two-area multimachine power system using input–output data without prior structural information.
- Comparison of three algorithms, PSO–QI, PSO and DEPSO, on system identification problems. Based on lower values of mean squared error and standard deviation, PSO–QI has shown to be the best algorithm compared to the other two for identification of IIR systems and generator dynamics in a multimachine power system.

The rest of the paper is organized as follows: In Section 2, an IIR system has been explained. The PSO–QI algorithm is explained in Section 3. In Section 4, results of studies carried out on some benchmark problems and a practical power system problem are presented. The conclusions are presented in Section 5.

2. An IIR system

System identification is the mathematical modeling of an unknown system by monitoring its input–output data. This is achieved by varying the parameters of the developed model so that for a set of given inputs, its output match that of the system under consideration. For a plant whose behavior is not known, an adaptive system can be modeled and its parameters can be continuously adjusted using any adaptive algorithms. By the use of such adaptive algorithms, the required parameters can be obtained such that the output of the plant and the model are same for the same set of inputs, which is the goal of system identification (Panda et al., 2007). Fig. 1 represents one such identification model of any arbitrary system.

As said, most nonlinear systems are also recursive in nature. Hence, models for real world systems are better represented as IIR systems. By doing so, the problem of system identification now becomes the problem of adaptive IIR filtering, for which different adaptive algorithms can be applied for adjusting the feed forward and feedback parameters of the recursive system. An IIR system can be represented by the transfer function:

$$H(z)=\frac{b_0+b_1z^{-1}+b_2z^{-2}+\cdots+b_mz^{-m}}{1+a_1z^{-1}+a_2z^{-2}+\cdots+a_nz^{-n}} \tag{1}$$

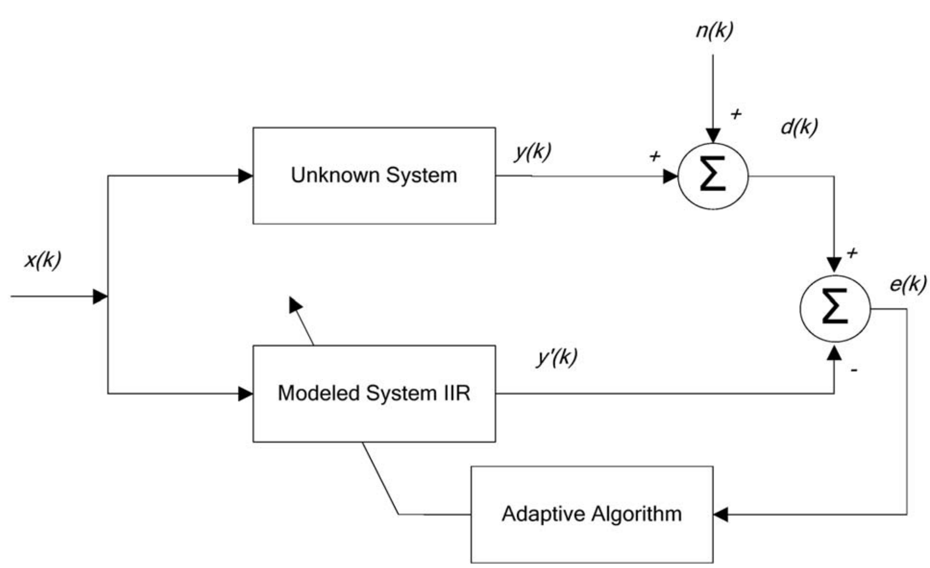

Fig. 1. Schematic for system identification.

where m and n are the number of numerator and denominator coefficients of the transfer function and a_n and b_m are the pole and zero parameters of the IIR filter. This can be written as difference equation of the form (Krusienski and Jenkins, 2005; Karaboga, 2005):

$$y(k) = \sum_{n=1}^{L} a_n(k)y(k-n) + \sum_{n=0}^{M} b_n(k)x(k-n) \quad (2)$$

where $x(k)$ and $y(k)$ represent the kth input and output of the system. Also, n=1, 2, 3,..., L and n=0, 1, 2,..., M represent the coefficients of the IIR filter. Considering the block diagram of Fig. 1, the output $y(k)$ for input $x(k)$ to the system is mixed with a noise signal $n(k)$. The output of the plant added with the noise gives the final system output $d(k)$. On the other hand, the output of the IIR filter in the modeled system for the same input $x(k)$ has an output of $y'(n)$. The difference of the output from the actual system with that of the modeled system gives the error $e(k)$. This error is used by the adaptive algorithm to adjust the parameters of the IIR filter, and thus reduce the error in a number of iterations so as to exactly identify the actual system. This has been shown in

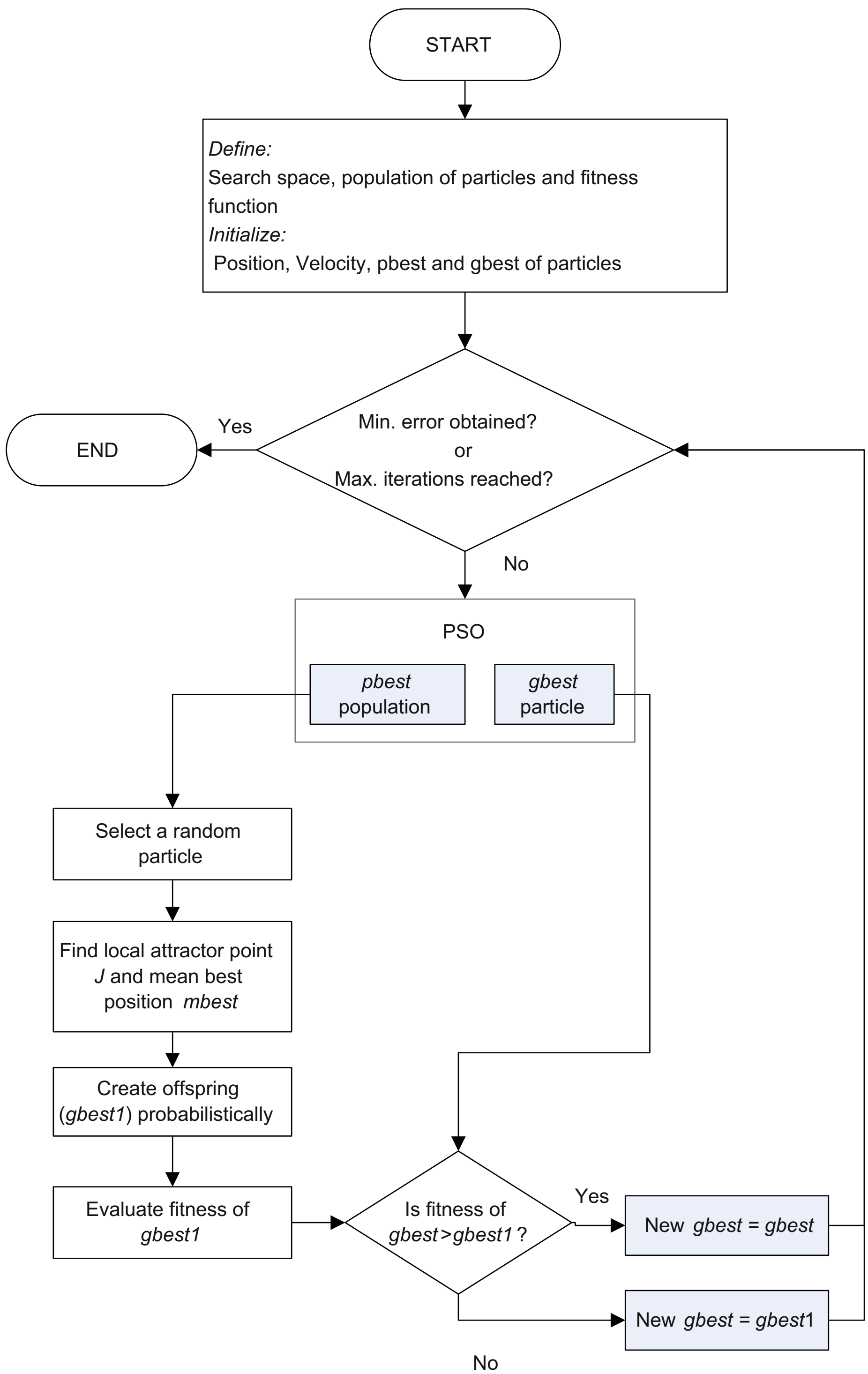

Fig. 2. Flowchart showing the PSO–QI algorithm.

Please cite this article as: Luitel, B., Venayagamoorthy, G.K., Particle swarm optimization with quantum infusion for system identification. Engineering Applications of Artificial Intelligence (2010), doi:10.1016/j.engappai.2010.01.022

the following equations:

$$d(k)=y(k)+n(k) \tag{3}$$

$$e(k)=d(k)-y'(k) \tag{4}$$

For the identification of the system, the adaptive algorithm tries to minimize the error $e(k)$ by adjusting the parameters of the modeled system, which are the pole-zero coefficients in case of an IIR system. The different kinds of algorithms that can be used for error minimization in adaptive systems are explained in Netto et al. (1995). In this paper, mean squared error (MSE) between the output of the actual system and the designed system as given by (5) has been considered as the feedback to the adaptive algorithm.

$$F=\frac{1}{N}\sum_{k=1}^{N}(d(k)-y'(k))^2 \tag{5}$$

The fitness function used by the different algorithms that are illustrated in the paper is given by

$$\text{Fitness}=\frac{1}{1+F} \tag{6}$$

The numerator and denominator coefficients of the IIR filter are represented by D dimensions (D=L+M). In Karaboga (2005), DE has been used for adjusting the parameters of the IIR system to reduce the MSE or to increase the fitness of the system. In this paper, PSO, DEPSO and PSO–QI are being used. The mentioned algorithms then find the best parameters in a number of iterations by searching for the possible solutions in a D-dimensional search space.

3. Particle swarm optimization with quantum infusion

Particle swarm optimization with quantum infusion is a new approach to hybridization of PSO and QPSO. Here, the quantum principle in QPSO is used to create a new offspring. After the positions and velocities of the particles are updated using standard PSO equations, a randomly chosen particle from PSO's *pbest* population is utilized to carry out the quantum operation (Luitel and Venayagamoorthy, 2009); and thus, create an offspring by mutating the *gbest*. The fitness of the offspring is evaluated and the offspring replaces the *gbest* particle of PSO only if it has a better fitness. This ensures that the fitness of the *gbest* particle is equal to or better than its fitness in the previous iteration. Thus, it is improved and pulled towards the best solution over iterations. By infusing the quantum theory to the standard PSO, a new hybrid algorithm is evolved which incorporates the best features of the respective individual algorithms and thus a better fitness is achieved. In PSO–QI, fast convergence property obtained by PSO in the first few iterations, and the convergence to a lower average error property obtained by QPSO, have been combined and hence the performance is significantly improved, as is shown in the results and Fig. 2. The flowchart for PSO–QI is illustrated in Fig. 2. It is described below in detail.

PSO is an evolutionary-like algorithm developed by Eberhart and Kennedy in 1995 (delValle et al., 2008). It is a population based search algorithm and is inspired by the observation of natural habits of bird flocking and fish schooling. In PSO, a swarm of particles moves through a D dimensional search space. The particles in the search process are the potential solutions, which move around a defined search space with some velocity until the error is minimized or the solution is reached, which is decided by the fitness function. The particles reach to the desired solution by updating their position and velocity according to the PSO equations. In PSO, each individual is treated as a volume-less particle in the D-dimensional space, with the position and velocity of the ith particle represented as

$$x_i=(x_{i1},x_{i2},\ldots,x_{iD}) \tag{7}$$

$$v_i=(v_{i1},v_{i2},\ldots,v_{iD}) \tag{8}$$

$$v_{id}(k+1)=w*v_{id}(k)+c_1*rand_1()*(P_{id}-x_{id})+c_2*rand_2()*(P_{gd}-x_{id}) \tag{9}$$

$$x_{id}(k+1)=x_{id}(k)+v_{id}(k+1) \tag{10}$$

These particles are randomly initialized over the search space with initial positions and velocities. They change their positions and velocities according to (9) and (10) where c_1 and c_2 are cognitive and social acceleration constants respectively, $rand_1()$

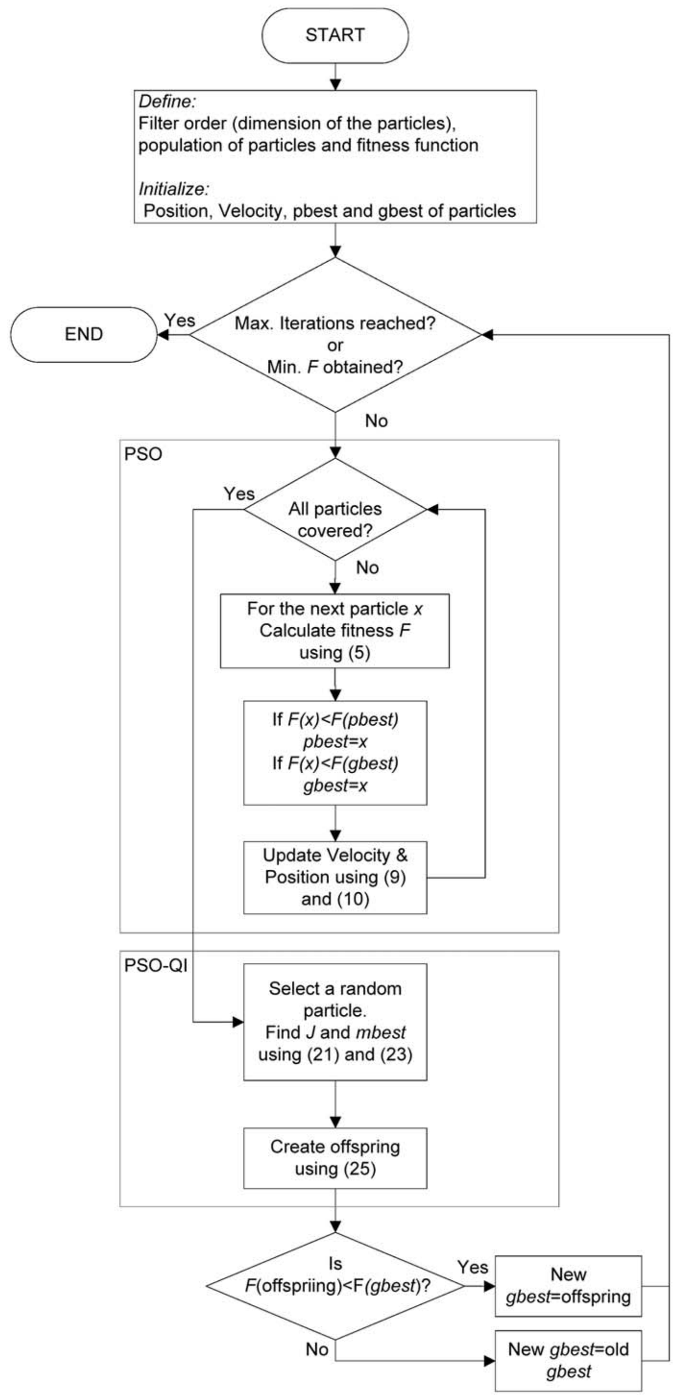

Fig. 3. Flowchart showing system identification using PSO–QI.

and $rand_2()$ are two random functions uniformly distributed in the range of [0,1] and w is the inertia weight introduced to accelerate the convergence speed of PSO (delValle et al., 2008). Vector $P_i=(P_{i1}, P_{i2},\ldots,P_{iD})$ is the best previous position (the position giving the best fitness value) of particle i called the *pbest*, and vector $P_g=(P_{g1}, P_{g2},\ldots, P_{gD})$ is the position of the best particle among all the particles in the swarm and is called the *gbest*. x_{id}, v_{id}, P_{id} are the dth dimension of vector of x_i, v_i, P_i. For the PSO equations to be dimensionally correct, the velocity term in (10) is taken over a unit time (Chakraborti et al., 2007). PSO is illustrated in the flowchart in Fig. 3.

Quantum behaved particle swarm optimization (QPSO) was introduced by Sun in 2004 (Sun et al., 2004a). According to the uncertainty principle, position and velocity of a particle in quantum world cannot be determined simultaneously. Thus QPSO differs from standard PSO mainly in the fact that exact values of x and v cannot be determined. In quantum mechanics, a particle, instead of having position and velocity, has a wavefunction given by

$$\psi(r,t) \tag{11}$$

which has no physical meaning but its amplitude squared gives the probability measure of its position in any one dimension r at time t. The governing equation of quantum mechanics is the Schrodinger's equation given by

$$j\hbar\frac{\partial}{\partial t}\psi(r,t)=\hat{H}(r)\psi(r,t) \tag{12}$$

where H is a time-independent Hamiltonian operator given by

$$\hat{H}(r)=-\frac{\hbar^2}{2m}\nabla^2+V(r) \tag{13}$$

where $\hbar$ is Planck's constant, m is the mass of the particle and $V_p(r)$ is the potential energy distribution (Mikki and Kishk, 2006). Based on the probability density function, a particle's probability of appearing in position x can be determined. Therefore in QPSO, a delta-potential-well based probability density function has been used with center of the well at point $J=(j_1, j_2,\ldots, j_D)$ in order to avoid explosion and help the particles in PSO to converge (Sun et al., 2004b). Assuming a particle in one-dimensional space having its center of potential at J, normalized probability density function Q and distribution function D_f can be obtained (Sun et al., 2005). Let $y=x-j$, then the form of this probability density function is given as follows and depends on the potential field the particle lies in:

$$Q(y)=\frac{1}{L}e^{-2|y|/L} \tag{14}$$

$$D_f(y)=\int_{-\infty}^{y}Q(y)\,dy=e^{-2|y|/L} \tag{15}$$

where the parameter L is the length of the potential field which depends on the energy intensity and is called the creativity or imagination of the particle that determines its search scope (Sun et al., 2004b). L can be evaluated as the distance between the particles' current position and point J as follows:

$$L=2\beta|J-x| \tag{16}$$

The parameter β is the only parameter of the algorithm. It is called the creativity coefficient and is responsible for the convergence speed of the particle. In QPSO, search and solution spaces are two unique spaces of different quality. So a mechanism is necessary to map the position of a particle in the search space to the solution space. This is referred to as 'collapsing' and is achieved by applying the Monte Carlo simulation. This is explained as follows (Sun et al., 2004a).

Let s be any random number uniformly distributed between 0 and $1/L$. For a uniform random number u in the interval [0,1], s is defined as

$$s=\frac{1}{L}u \tag{17}$$

Now, equating (14) and (17), the following relation is achieved:

$$u=e^{-2|y|/L} \tag{18}$$

$$y=\pm\frac{L}{2}\ln\left(\frac{1}{u}\right) \tag{19}$$

The position equation is given as follows:

$$x=J\pm\frac{L}{2}\ln\left(\frac{1}{u}\right) \tag{20}$$

where the particle's local attractor point J has coordinates given by the following equation:

$$J_d=\alpha_1P_{gd}+\alpha_2P_{id} \tag{21}$$

where $\alpha_1=a/(a+b)$ and $\alpha_2=b/(a+b)$, and a and b are two uniformly distributed random numbers.

From (16) and (19), the new position of the particle is calculated as

$$x(k+1)=J(k)\pm\beta|J(k)-x(k)|\ln\left(\frac{1}{u}\right) \tag{22}$$

This delta-potential-well based quantum PSO is called the QDPSO in Sun et al. (2004a). This has been improved further by defining a mainstream thought (Sun et al., 2005) or the mean best

Table 1
Study of Cases I and II.

			Case I (Krusienski and Jenkins, 2004)	Case II (Ng et al., 1996)
Transfer function			$\frac{1.25z^{-1}-0.25z^{-2}}{1-0.3z^{-1}+0.4z^{-2}}$	$\frac{-0.2z^{-1}-0.4z^{-2}+0.5z^{-3}}{1+0.6z^{-1}-0.25z^{-2}+0.2z^{-3}}$
Full order	L		2	3
	M		1	2
	Model		$\frac{b_0+b_1z^{-1}}{1+a_1z^{-1}+a_2z^{-2}}$	$\frac{b_0+b_1z^{-1}+b_2z^{-2}}{1+a_1z^{-1}+a_2z^{-2}+a_3z^{-3}}$
Reduced order	L		1	2
	M		0	1
	Model		$\frac{b_0}{1+a_1z^{-1}}$	$\frac{b_0+b_1z^{-1}}{1+a_1z^{-1}+a_2z^{-2}}$

Please cite this article as: Luitel, B., Venayagamoorthy, G.K., Particle swarm optimization with quantum infusion for system identification. Engineering Applications of Artificial Intelligence (2010), doi:10.1016/j.engappai.2010.01.022

ARTICLE IN PRESS

position, *mbest*, as

$$mbest(k) = \frac{1}{S}\sum_{i=1}^{S} P_i(k) = \left(\frac{1}{S}\sum_{i=1}^{S} P_{i1}(k), \ldots, \frac{1}{S}\sum_{i=1}^{S} P_{iD}(k)\right) \quad (23)$$

where S is the size of the population, D is the number of dimensions and P_i is the *pbest* position of each particle. Now the position update equation in (22) is given as (24), where the addition or subtraction is carried out with 50% probability:

$$x(k+1) = J(k) \pm \beta\,|mbest(k) - x(k)|\,\ln\left(\frac{1}{u}\right) \quad (24)$$

By using (21) this can also be written as follows to show the mutation on *gbest*:

$$x(k+1) = \alpha_1 P_{gd}(k) + \alpha_2 P_{id}(k) \pm \beta\,|mbest(k) - x(k)|\,\ln\left(\frac{1}{u}\right) \quad (25)$$

4. Case studies and results

Two different studies have been carried out for system identification. In the first study, four benchmark IIR systems between second and sixth order are considered for the case study.

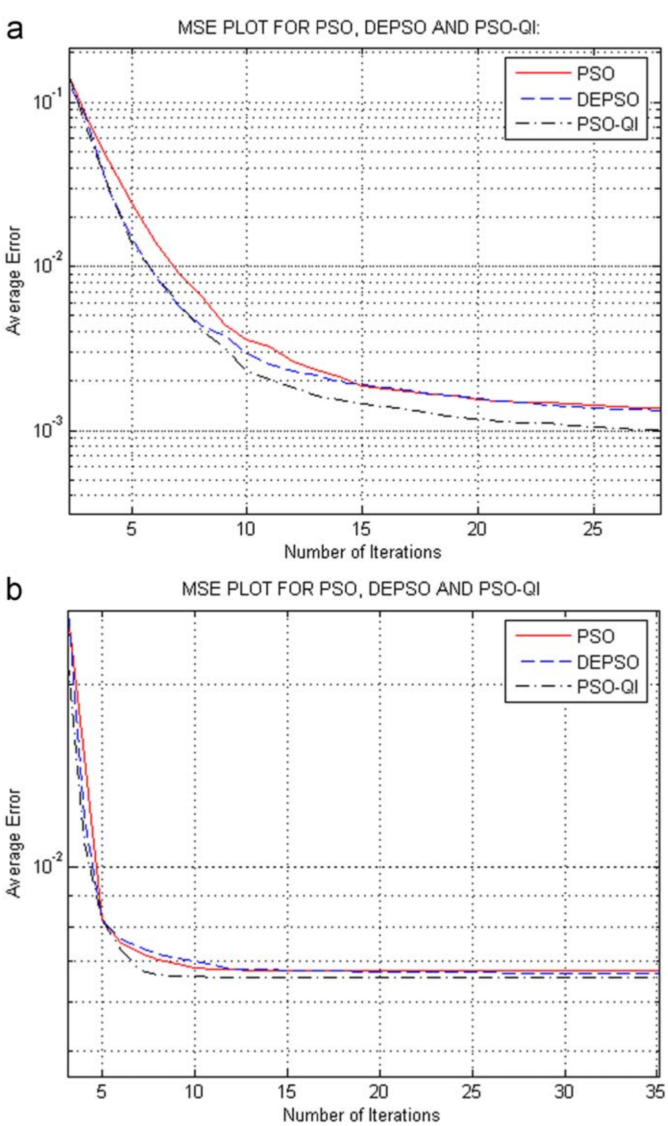

Fig. 4. Error graph for Case I: (a) full order and (b) reduced order for 500 iterations.

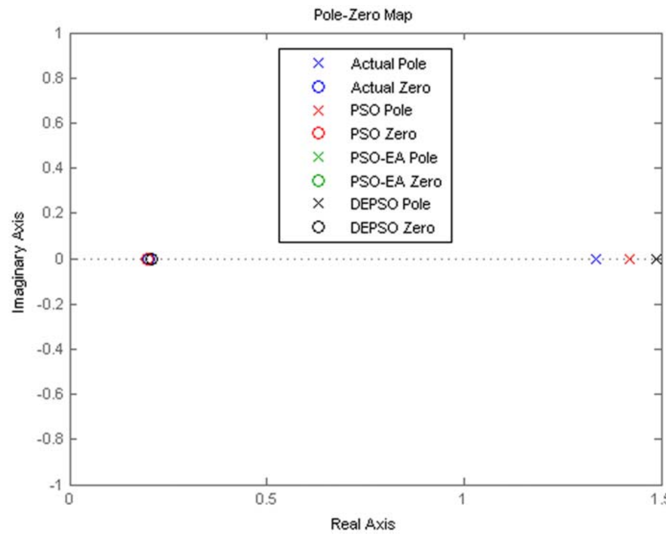

Fig. 5. Pole zero plot for full order model of Case I.

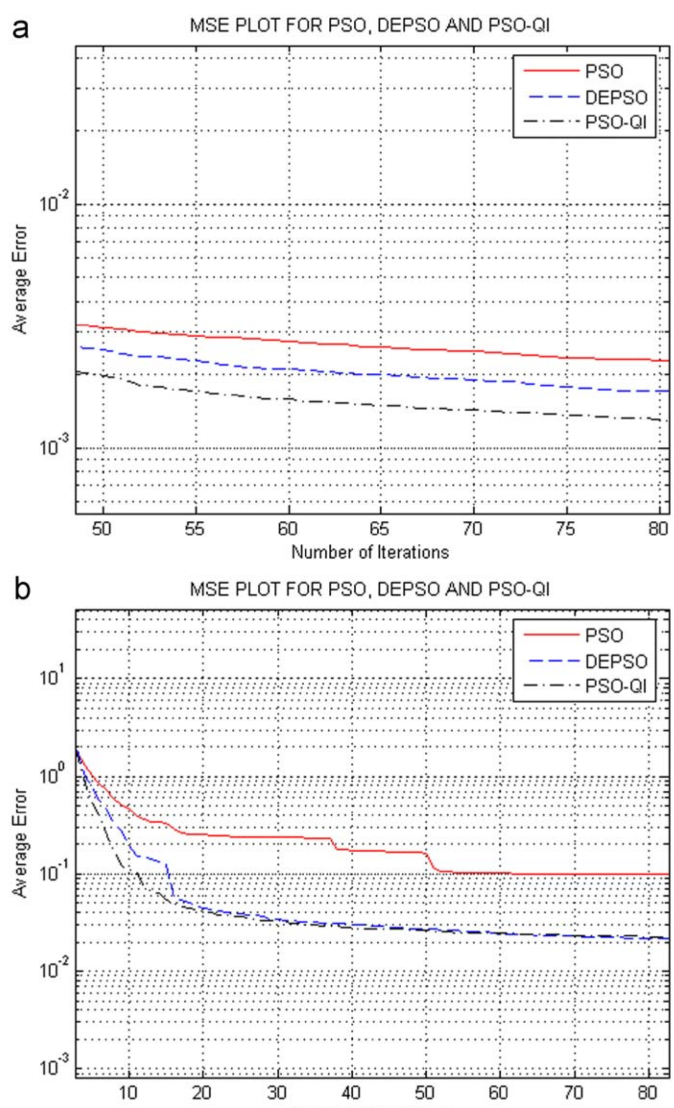

Fig. 6. Error graph for Case II: (a) full order and (b) reduced order for 500 iterations.

Please cite this article as: Luitel, B., Venayagamoorthy, G.K., Particle swarm optimization with quantum infusion for system identification. Engineering Applications of Artificial Intelligence (2010), doi:10.1016/j.engappai.2010.01.022

These IIR systems are modeled using two different models, one having the same order as the actual system and second having less order than that of the actual system. These reduced order cases pose challenge to the optimization algorithm in that they produce multimodal error surface which has multiple minima. In all cases, as the number of coefficients decreases, the degree of freedom reduces and it becomes more difficult to identify the actual system. In the second study, identification of dynamics of four generators in a power system is considered. With pseudo-random binary signal (PRBS) input to one of the generators, the speed deviations on all of the four generators is measured. From this input–output data, the transfer function of the four generators for dynamics of interest is identified using PSO, DEPSO and PSO–QI. Identification of IIR systems using PSO–QI is shown in the flowchart in Fig. 3.

4.1. Study I

Each case is simulated using PSO, DEPSO (Luitel and Venayagamoorthy, 2008a) and PSO–QI (Luitel and Venayagamoorthy, 2008b) in MATLAB on the same computer using the following parameters. The PSO parameters used in the study are obtained by systematic study of the effect of various parameters in different case studies. The variation of the results with the parameters is, however, not a part of the results shown in this paper. The shown result is an average over 50 trials.

D=number of dimension representing the weights to be optimized

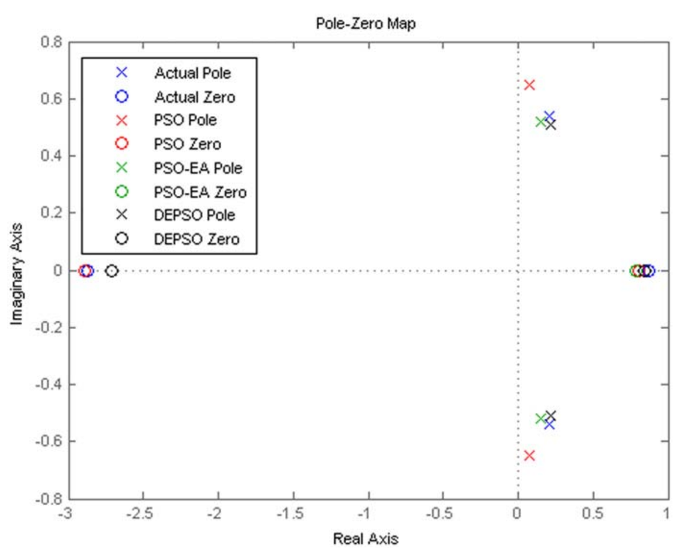

Fig. 7. Pole-zero plot for full order model of Case II.

Table 2
Study of Case III (Shynk, 1989b).

Transfer function		$\frac{z^{-1}-0.9z^{-2}+0.81z^{-3}-0.729z^{-4}}{1-0.04z^{-1}-0.2775z^{-2}+0.2101z^{-3}-0.14z^{-4}}$
Full order	L	4
	M	3
	Model	$\frac{b_0+b_1z^{-1}+b_2z^{-2}+b_3z^{-3}}{1+a_1z^{-1}+a_2z^{-2}+a_3z^{-3}+a_4z^{-4}}$
Reduced order	L	3
	M	2
	Model	$\frac{b_0+b_1z^{-1}+b_2z^{-2}}{1+a_1z^{-1}+a_2z^{-2}+a_3z^{-3}}$

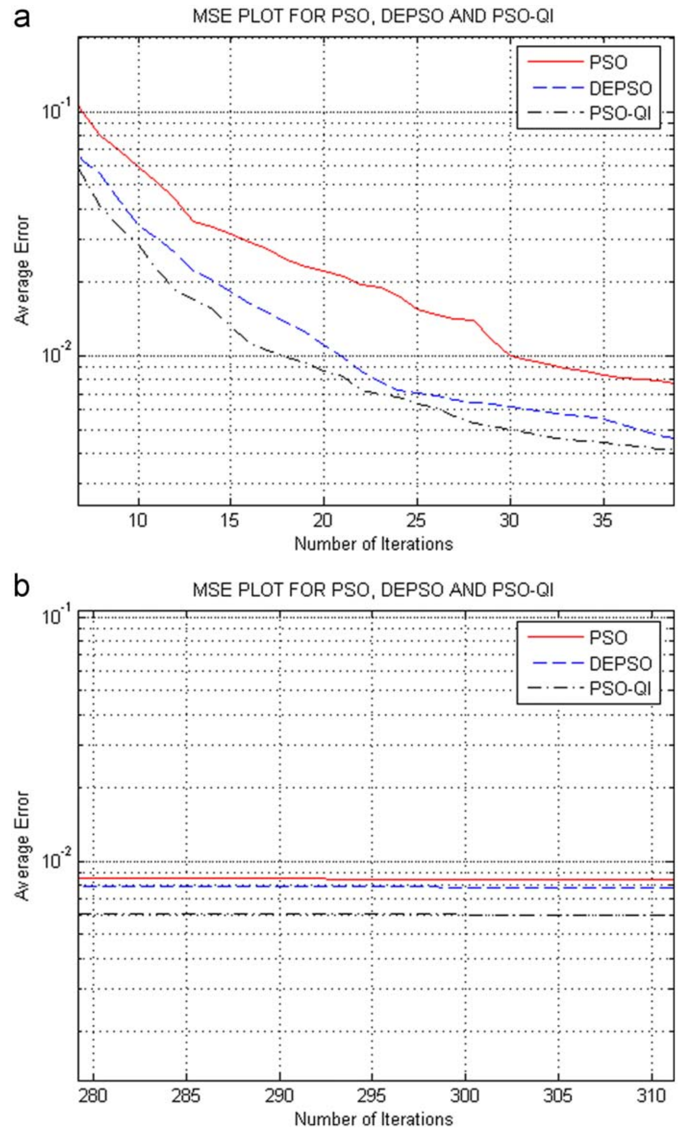

Fig. 8. Error graph for Case III: (a) full order and (b) reduced order for 500 iterations.

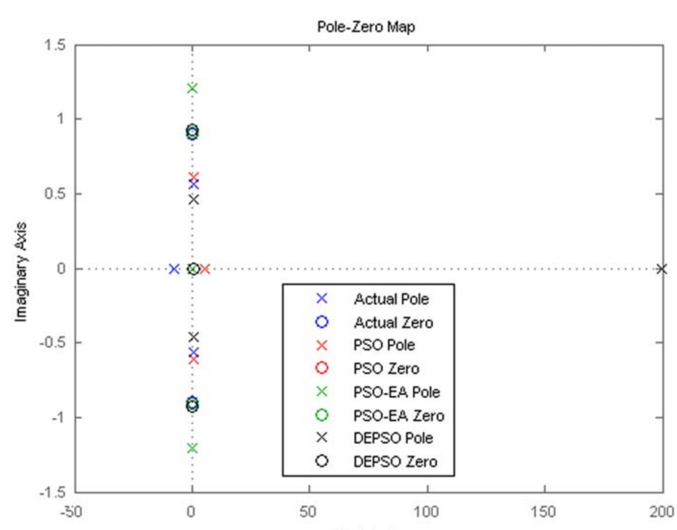

Fig. 9. Pole-zero plot for full order model of Case III.

Please cite this article as: Luitel, B., Venayagamoorthy, G.K., Particle swarm optimization with quantum infusion for system identification. Engineering Applications of Artificial Intelligence (2010), doi:10.1016/j.engappai.2010.01.022

Table 3
Study of Case IV (Karaboga, 2005).

Transfer function	$\frac{1-0.4z^{-2}-0.65z^{-4}+0.26z^{-6}}{1-0.77z^{-2}-0.8498z^{-4}+0.6486z^{-6}}$	
Full order	L	6
	M	6
	Model	$\frac{b_0+b_1z^{-1}+b_2z^{-2}+b_3z^{-3}+b_4z^{-4}+b_5z^{-5}+b_6z^{-6}}{1+a_1z^{-1}+a_2z^{-2}+a_3z^{-3}+a_4z^{-4}+a_5z^{-5}+a_6z^{-6}}$
Reduced order	L	5
	M	5
	Model	$\frac{b_0+b_1z^{-1}+b_2z^{-2}+b_3z^{-3}+b_4z^{-4}+b_5z^{-5}}{1+a_1z^{-1}+a_2z^{-2}+a_3z^{-3}+a_4z^{-4}+a_5z^{-5}}$

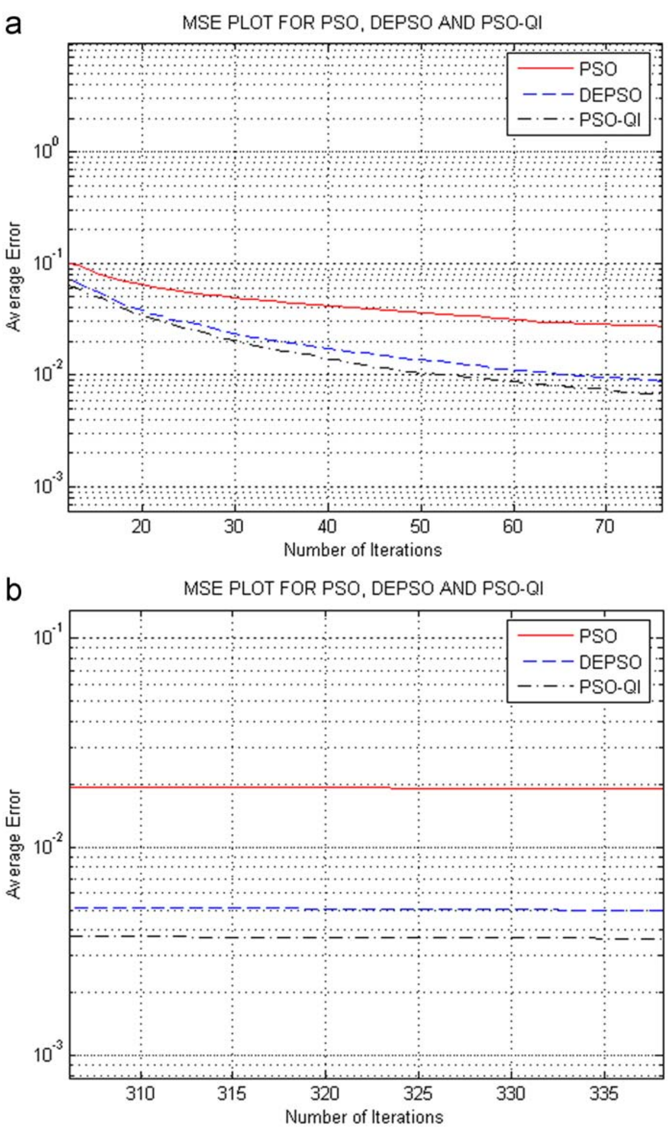

Fig. 10. Error graph for Case IV: (a) full order and (b) reduced order for 500 iterations.

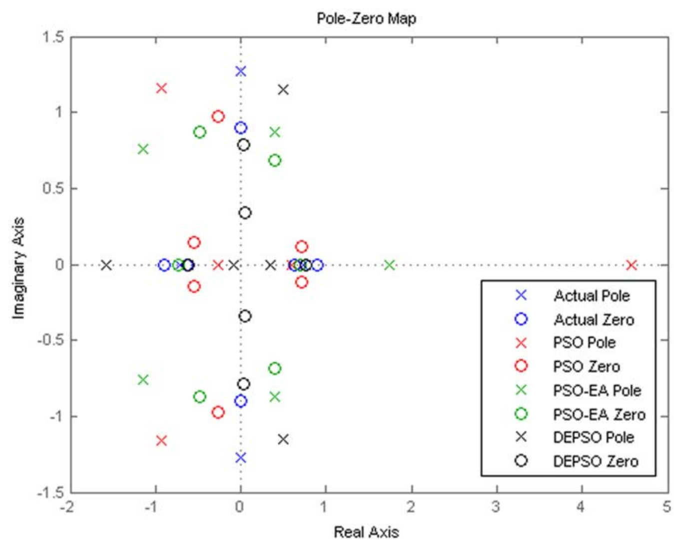

Fig. 11. Pole-zero plot for full order model of Case IV.

P (population size)=25
c_1, c_2 (cognitive and social acceleration constants for PSO)=2
w (inertia weight)=linearly decreasing from 1.4 to 0
CR (crossover rate)=0.8
β=linearly increasing from 0.5 to 1
Number of inputs=50
Number of iterations=500
Maximum velocity=1.3
Maximum position=1.3
Number of trials=50

The coefficients are randomly initialized within the periphery of the possible solution and the maximum velocity and position are also restricted to 1.3 which is the maximum value of the actual coefficients of the plant. The output of the plant is subjected to a white Gaussian noise of -30 dB signal to noise ratio.

The transfer functions and their implementation in two different models for Cases I and II are shown in Table 1. Case I is a second order system. The simulation results of two different models for this case are shown in Fig. 4(a) and (b). The pole-zero plot of the coefficients obtained for this transfer function is shown in Fig. 5. Fig. 6(a) and (b) show the results of the third order system studied in Case 2. Fig. 7 shows the pole-zero plot of the coefficients obtained for the transfer function. Case 3 is a fourth order IIR system shown in Table 2. The error graphs for the two different models are shown in Fig. 8(a) and (b). The pole-zero plot for this case is shown in Fig. 9. Table 3 shows the transfer function of the plant and its models for the sixth order system studied in Case 4. The results are shown in Fig. 10(a) and (b) and the pole-zero plot in Fig. 11.

The comparison of performance of PSO, DEPSO and PSO–QI is shown in Table 4 where minimum, average and standard deviation of the results obtained from 50 trials over 500 iterations have been presented. These results show that

Table 4
Full order model (500 iterations).

Case		MSE (dB)			Time (s)[a]	
		Min.	Avg.	Std.	Min.	Avg.
Case I	PSO	7.102e – 4	8.612e – 4	5.074e – 4	3.422	3.769
	DEPSO	7.102e – 4	7.278e-4	4.391e – 5	2.547	3.166
	PSO–QI	7.102e – 4	**7.102e – 4**	**1.148e – 7**	2.984	3.227
Case II	PSO	7.791e – 4	0.001	5.222e – 4	3.563	3.778
	DEPSO	7.791e – 4	9.480e – 4	4.011e – 4	2.703	2.826
	PSO–QI	7.791e – 4	**9.215e – 4**	**3.627e – 4**	3.281	3.432
Case III	PSO	7.245e – 4	0.003	0.003	2.609	3.404
	DEPSO	7.245e – 4	0.001	0.001	2.672	3.056
	PSO–QI	7.245e – 4	**0.001**	**0.001**	3.421	3.734
Case IV	PSO	7.821e – 4	0.011	0.014	0.938	2.240
	DEPSO	7.623e – 4	0.002	0.003	1.046	2.329
	PSO–QI	7.984e – 4	**0.002**	**0.004**	3.063	4.008

[a] Performed on the same computer for 500 iterations.

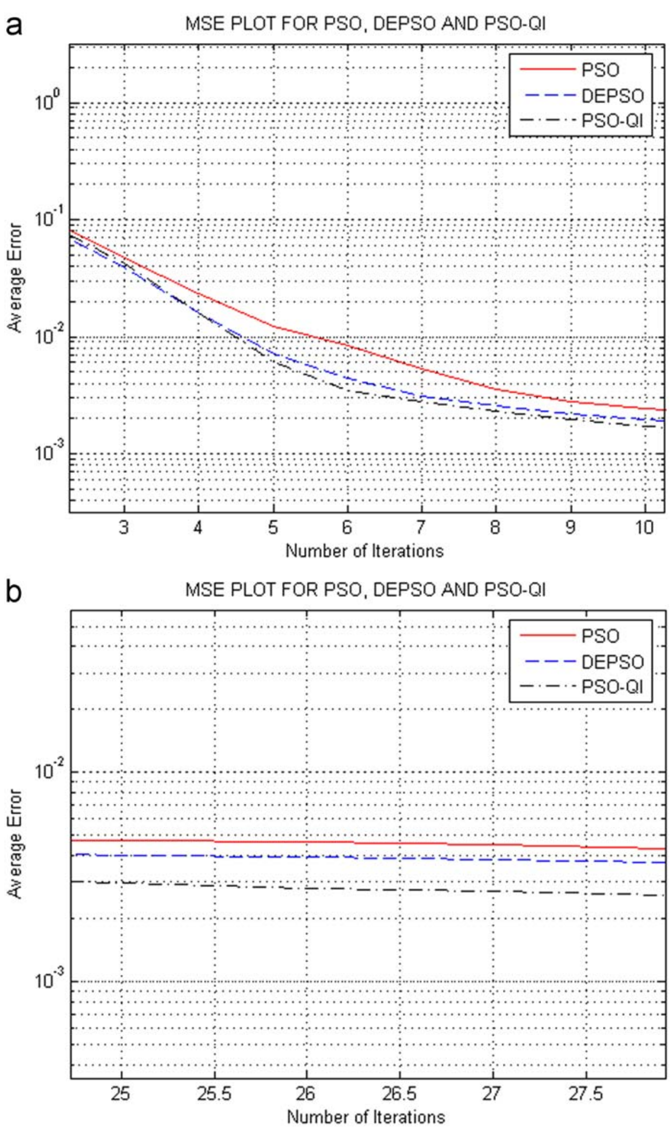

Fig. 12. Error graph for: (a) Case I and (b) Case II for 50 iterations.

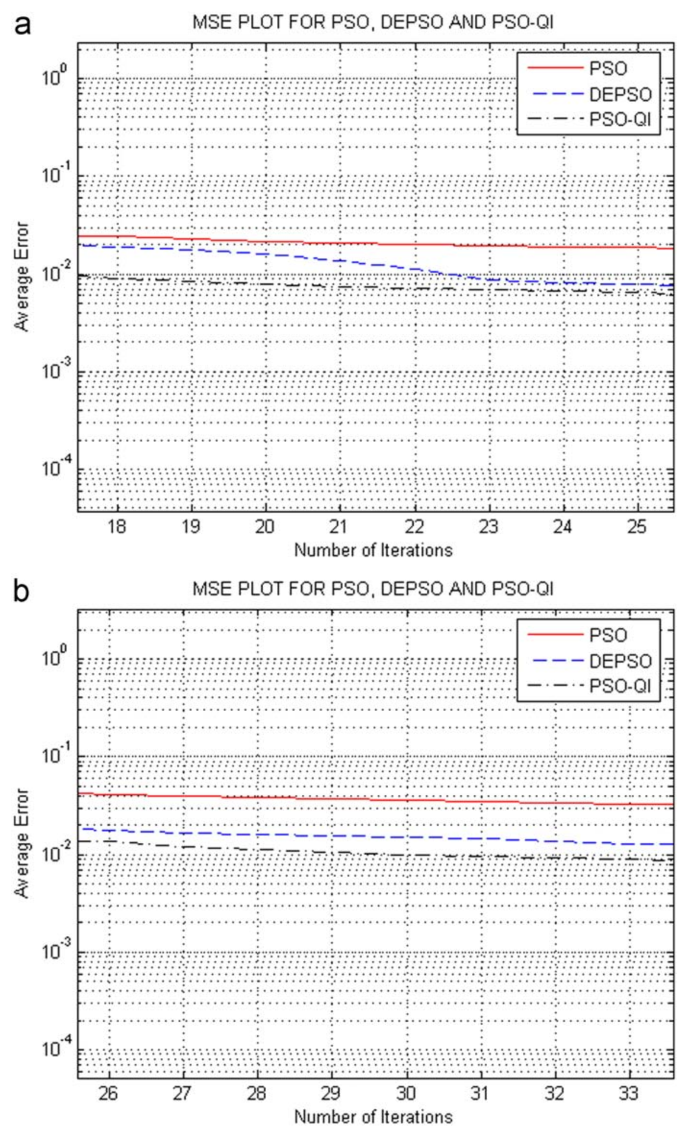

Fig. 13. Error graph for: (a) Case III and (b) Case IV for 50 iterations.

Please cite this article as: Luitel, B., Venayagamoorthy, G.K., Particle swarm optimization with quantum infusion for system identification. Engineering Applications of Artificial Intelligence (2010), doi:10.1016/j.engappai.2010.01.022

although PSO–QI takes slightly more time to converge to the global minimum, it converges to a lower MSE. Moreover, the results show that PSO–QI is fairly consistent in its performance and it deviates the least over a number of trials. The study is also carried out for full order with 50 iterations. These results for Cases I and II are shown in Fig. 12(a) and (b) and for Cases III and IV in Fig. 13(a) and (b), respectively. Table 5 shows the comparison of the performance of the three algorithms in 50 iterations. These results also confirm the robustness of PSO–QI in its ability to converge faster and to a higher fitness value. The similar results for the reduced order case are shown in Table 6. These results also indicate the better performance of PSO–QI over PSO and DEPSO. Since mutation operation introduced by DE helps the DEPSO algorithm to come out of the local minima, it finds the global minimum faster, where PSO tends to get stuck. However, PSO–QI has even better ability to overcome the local minima due to its quantum operation based mutation on the global best particle obtained from PSO.

4.2. Study II

In this study, identification of generator dynamics in a power system is carried out based on its input–output data with no prior information about the structure of the system. The system is implemented using (26) where $x(k)$ and $y(k)$ are the input and output samples at time k. $\hat{Y}(k)$ represents the output obtained by the designed system at instant k

$$\hat{Y}(k) = a_1x(k) + a_2x(k-1) + a_3x(k-2) + b_1y(k-1) + b_2y(k-2) + b_3y(k-3) \tag{26}$$

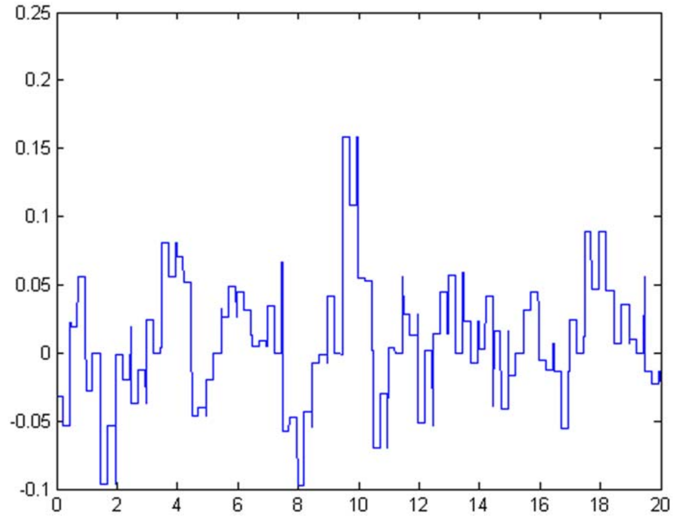

Fig. 14. Input pseudo-random binary signal.

Table 5
Full order model (50 iterations).

Case		MSE (dB)			Time (s)[a]	
		Min.	Avg.	Std.	Min.	Avg.
Case I	PSO	9.448e−4	0.001	5.011e−4	0.218	0.261
	DEPSO	9.448e−4	0.001	5.806e−4	0.234	0.302
	PSO–QI	9.447e−4	**9.988e−4**	**1.222e−4**	0.265	0.275
Case II	PSO	0.001	0.002	0.003	0.234	0.264
	DEPSO	0.001	0.002	0.003	0.235	0.263
	PSO–QI	0.001	**0.001**	**5.674e−4**	0.297	0.343
Case III	PSO	0.001	0.013	0.045	0.233	0.269
	DEPSO	0.001	0.004	0.002	0.250	0.275
	PSO–QI	8.353e−4	**0.003**	**0.002**	0.343	0.371
Case IV	PSO	0.001	0.024	0.032	0.234	0.257
	DEPSO	8.688e−4	0.007	0.010	0.250	0.267
	PSO–QI	9.994e−4	**0.004**	**0.006**	0.375	0.399

[a] Performed on the same computer for 500 iterations.

Table 6
Reduced order model (500 iterations).

Case		MSE (dB)			Time (s)[a]	
		Min.	Avg.	Std.	Min.	Avg.
Case I	PSO	0.006	0.006	7.149e−4	2.234	2.356
	DEPSO	0.006	0.006	4.214e−4	2.125	2.326
	PSO–QI	0.006	**0.006**	**4.085e−18**	2.500	2.601
Case II	PSO	0.004	0.089	0.443	3.625	3.799
	DEPSO	0.004	0.010	0.005	3.609	3.700
	PSO–QI	0.004	**0.011**	**0.006**	3.079	3.130
Case III	PSO	0.005	0.008	0.003	0.766	1.269
	DEPSO	0.005	0.007	0.003	0.859	1.392
	PSO–QI	0.005	**0.005**	**0.001**	2.312	2.700
Case IV	PSO	0.001	0.018	0.042	2.281	2.766
	DEPSO	0.001	0.004	0.004	2.515	2.678
	PSO–QI	0.001	**0.003**	**0.001**	3.375	3.627

[a] Performed on the same computer for 500 iterations.

Please cite this article as: Luitel, B., Venayagamoorthy, G.K., Particle swarm optimization with quantum infusion for system identification. Engineering Applications of Artificial Intelligence (2010), doi:10.1016/j.engappai.2010.01.022

This can be written into a transfer function as follows:

$$\frac{Y(z)}{X(z)} = \frac{a_1 + a_2 z^{-1} + a_3 z^{-2}}{1 - b_1 z^{-1} - b_2 z^{-2} - b_3 z^{-3}} \quad (27)$$

$$e(k) = y(k) - \hat{Y}(k) \quad (28)$$

After filtering the input and the delayed output data, the output of the system is compared with the actual output and the MSE between these outputs is taken as the fitness function. Using the fitness information, the algorithms continuously adjust the coefficients of the filters so as to minimize the error between the actual output of the system and the output of the designed system given by (28). In this study, four generators (G1–G4) of a two-area power system (Venayagamoorthy, 2007) are considered. The generator G1 is subjected to a PRBS input and speed deviation (dSpeed) of the four generators is recorded for 20 s of input data (3125 samples). The first 10 s of data is taken for identification of the system (training) and the next 10 s of data is used to verify the

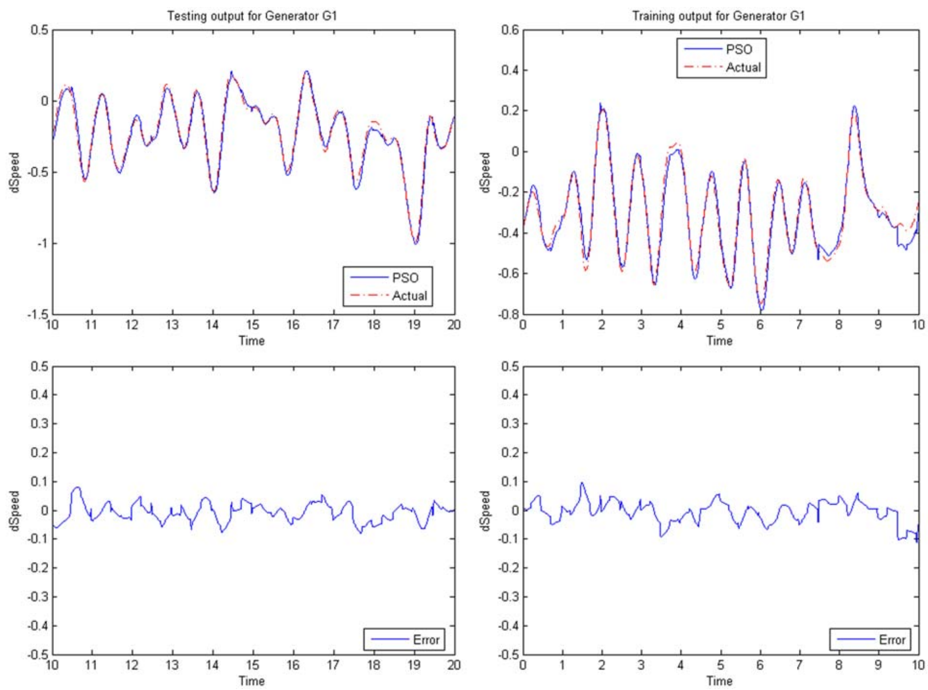

Fig. 15. Training and testing plots for G1 using PSO.

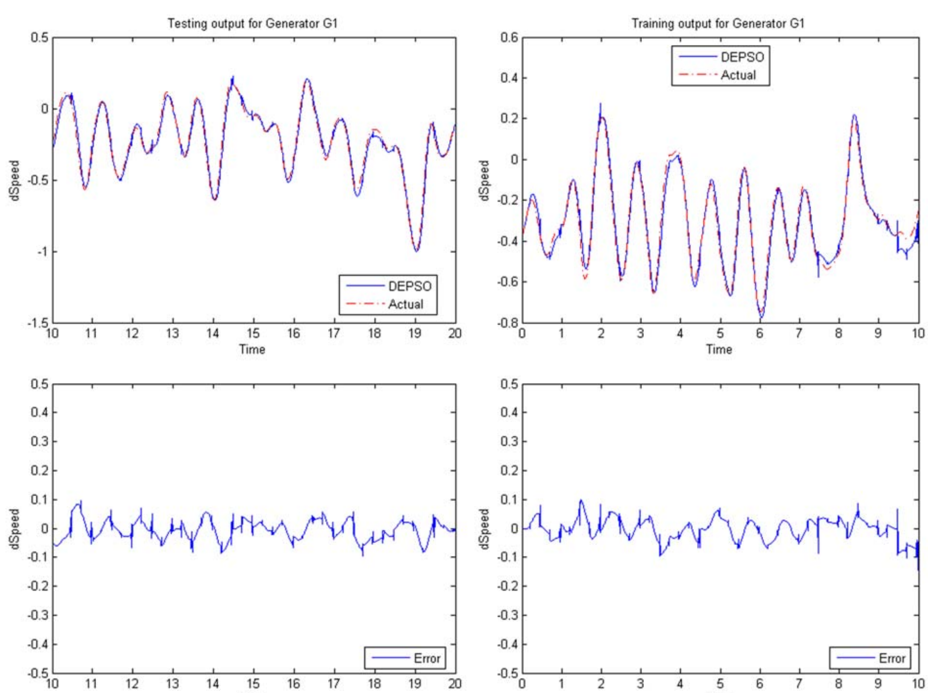

Fig. 16. Training and testing plots for G1 using DEPSO.

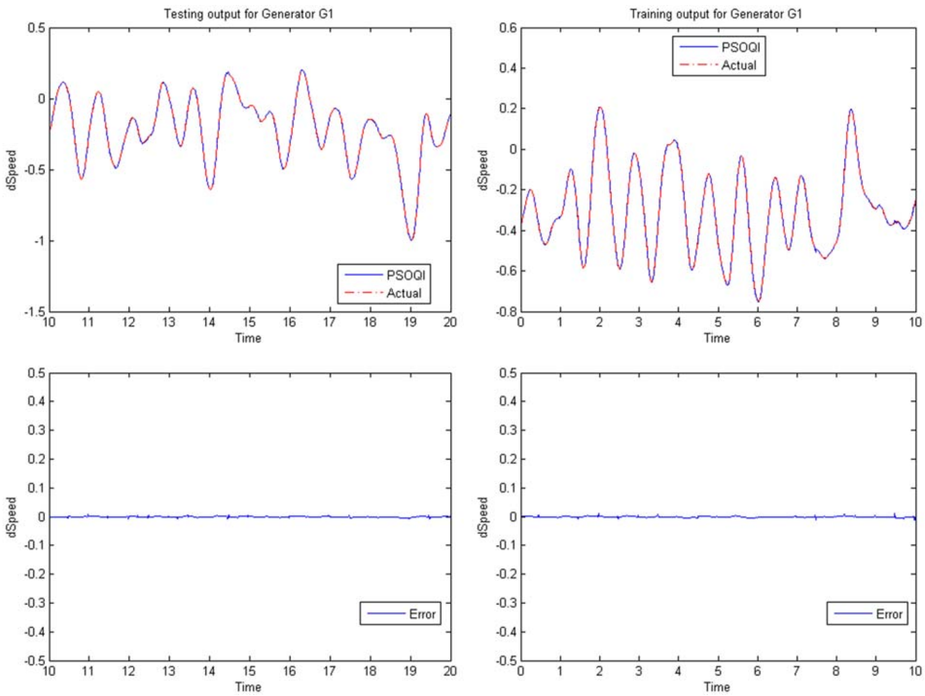

Fig. 17. Training and testing plots for G1 using PSO–QI.

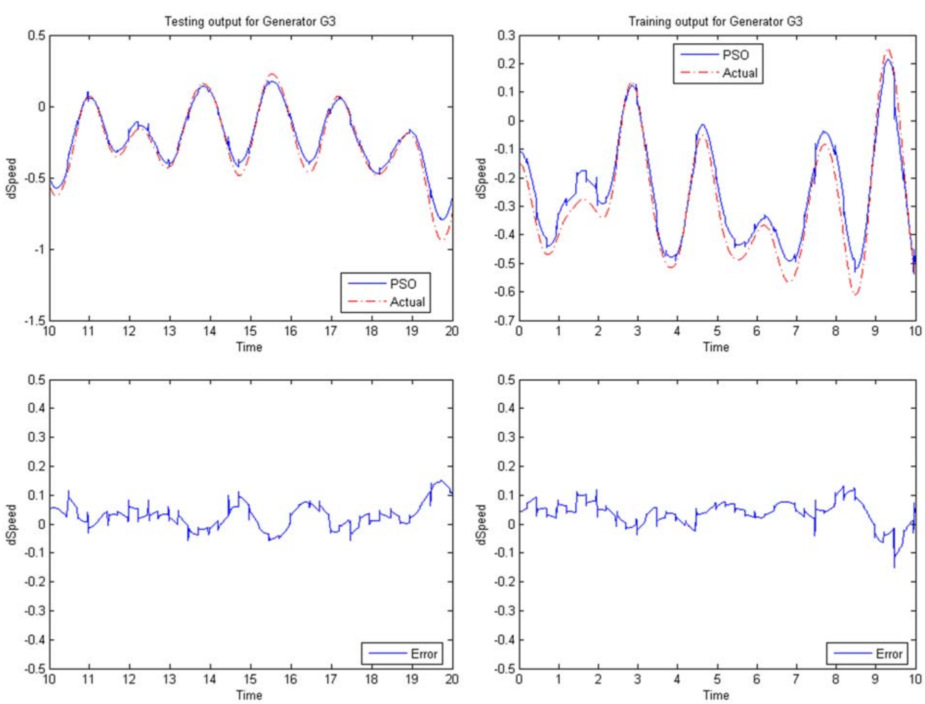

Fig. 18. Training and testing plots for G3 using PSO.

system output (testing). The training and testing plots for different generators are shown in Figs. 14–21. In this study, the dimension of the system is 6, each dimension representing the coefficient of the transfer function.

The input PRBS signal is shown in Fig. 14. The training and testing plot for G1 obtained using PSO, DEPSO and PSO–QI are presented in Figs. 15–17, respectively. Figs. 18–20 show the training and testing plots for G3 obtained using PSO, DEPSO and PSO–QI, respectively. Units in Figs. 15–20 for dspeed and time are rad/s and s respectively. The minimum and average values of fitness for the three algorithms obtained from the study are presented

in Table 7. The standard deviation of the minimum values over 50 trials is also presented in the table. The results show that PSO–QI is more consistent in identifying the dynamics of the system in every trial. Although PSO and DEPSO could also identify the system transfer function and predict the speed deviation, they were not able to do so in every trial. PSO and

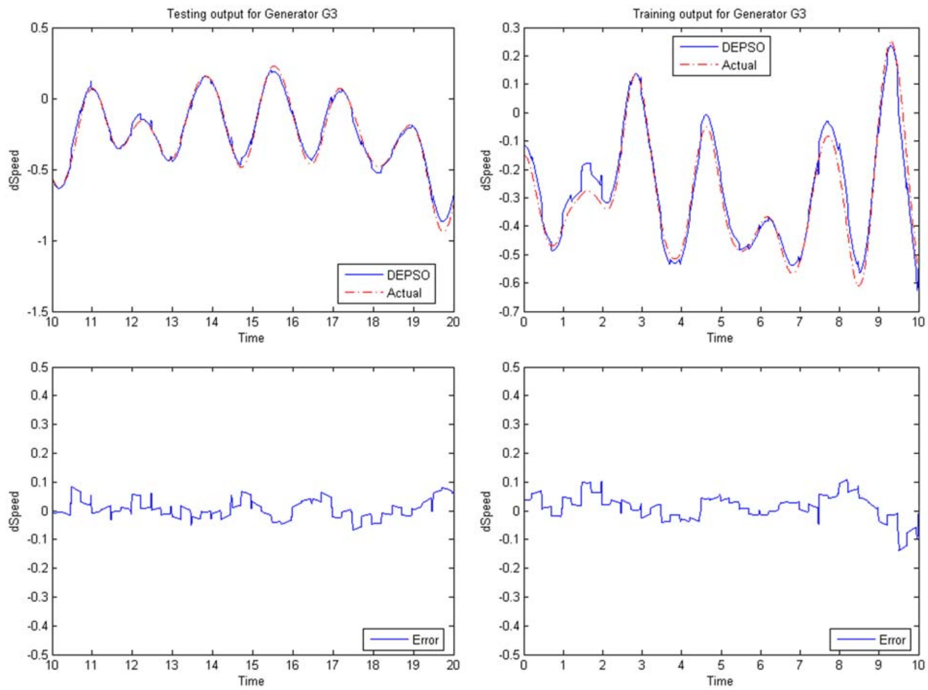

Fig. 19. Training and testing plots for G3 using DEPSO.

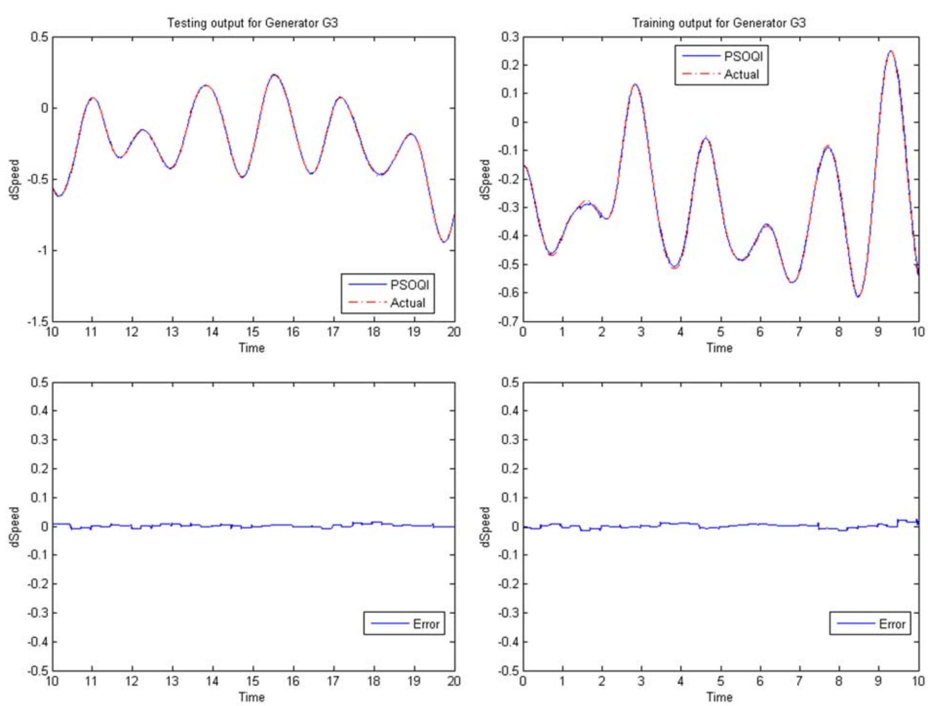

Fig. 20. Training and testing plots for G3 using PSO–QI.

DEPSO also suffered from the problem of identifying an out of phase system.

Although the fitness obtained by PSO–QI is better than PSO, it has to go through more number of fitness evaluations. Each iteration of PSO corresponds to P fitness evaluations while that of PSO–QI corresponds to $P+1$ fitness evaluations. However, if more number of *pbest* particles is considered for quantum operation, the fitness evaluations will increase. In these studies, all of the *pbest* particles are considered for quantum operation and hence the number of fitness evaluations per iteration is $2P$. However, given the equal number of fitness evaluations, standard PSO does not show improvements in fitness as is shown in Fig. 21. The figure also demonstrates how the PSO–QI converges to a lower fitness in less number of iterations, thus showing its promises in online applications, and how the fitness of PSO does not meet that of PSO–QI regardless of the number of fitness evaluations (proportional to the number of iterations).

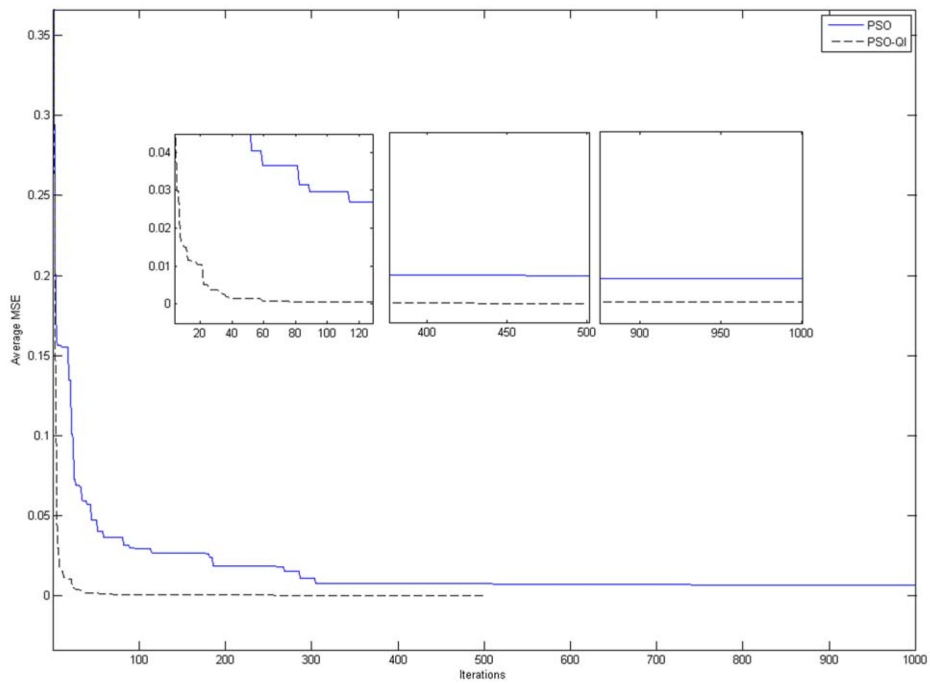

Fig. 21. Comparison of PSO and PSO–QI in terms of fitness.

Table 7
Comparison of PSO, DEPSO and PSO–QI for system identification.

		PSO		DEPSO		PSO–QI	
		Training	Testing	Training	Testing	Training	Testing
G1	Min.	0.0288	0.0231	0.0090	0.0090	0.0014	0.0011
	Avg.	0.3558	0.2790	0.3859	0.3028	**0.0098**	**0.0084**
	Std.	0.2610	0.2046	0.2828	0.2212	**0.0058**	**0.0054**
G2	Min.	0.0576	0.0492	0.0075	0.0057	8.0904e-4	7.0726e-4
	Avg.	0.3843	0.3219	0.2675	0.2237	**0.0085**	**0.0071**
	Std.	0.2463	0.2079	0.2386	0.2011	**0.0057**	**0.0046**
G3	Min.	0.0126	0.0127	0.0024	0.0021	0.0020	0.0022
	Avg.	0.3417	0.3156	0.2183	0.1973	**0.0200**	**0.0190**
	Std.	0.2528	0.2365	0.2553	0.2369	**0.0773**	**0.0726**
G4	Min.	0.0272	0.0256	0.0025	0.0026	7.8092e-4	7.7778e-4
	Avg.	0.3394	0.3082	0.2807	0.2543	**0.0089**	**0.0086**
	Std.	0.2356	0.2156	0.3020	0.2758	**0.0120**	**0.0111**

5. Conclusion

A hybrid particle swarm optimization with quantum infusion (PSO–QI) algorithm for identification of IIR systems and generator dynamics in a multimachine power system has been presented in this paper. The performance of PSO–QI is compared with that of PSO and a hybrid algorithm of PSO and differential evolution (DEPSO). These studies performed show that swarm, evolutionary and quantum behaved algorithms can be applied in system identification and hybrid algorithms perform better by combining the best features of the participating individual algorithms. The lower values of mean squared error and standard deviation show that PSO–QI is the best algorithm among the three for system identification. The results show that PSO–QI converges faster and with more consistency than the other algorithms, thus showing its promise in online implementation. However, it is computationally complex due to the increased number of fitness evaluations and hence a trade-off between time complexity and fitness is necessary.

To confirm its robustness and scalability, PSO–QI needs to be applied to many different benchmark problems and dynamical real world applications. For the authors' future work, the algorithm will be tested on different types of applications, including online and hardware implementation, using different fitness functions.

Acknowledgments

Funding provided by the National Science Foundation, USA under the CAREER grant ECCS #0348221 and EFRI #0836017 is gratefully acknowledged.

References

Chakraborti, N., Das, S., Jaykanth, R., Pekoz, R., Erkoc, S., 2007. Genetic algorithms applied to Li+ ions contained in carbon nanotubes: an investigation using particle swarm optimization and differential evolution along with molecular dynamics. Materials and Manufacturing Processes 22, 562–569.

Chen, S., Mei, T., Luo, M.,Yang, X., Identification of nonlinear system based on a new hybrid gradient-based PSO algorithm. In: International Conference on Information Acquisition, July 2007, pp. 265–268.

delValle, Y., Venayagamoorthy, G.K., Mohagheghi, S., Hernandez, J.C., Harley, R.G., 2008. Particle swarm optimization: basic concepts, variants and applications in power systems. IEEE Transactions on Evolutionary Computation 12, 171–195.

Hongwei, G., Yanchun L., Identification for nonlinear systems based on particle swarm optimization and recurrent neural network [ultrasonic motor control applications]. In: Proceedings of the International Conference on Communications, Circuits and Systems, vol. 2, May 2005.

Karaboga, N., 2005. Digital IIR filter design using differential evolution algorithm. EURASIP Journal of Applied Signal Processing 8, 1269–1276.

Karaboga, N., Kalinli, A., Karaboga, D., 2004. Designing digital IIR filters using ant colony optimization algorithm. Engineering Applications of Artificial Intelligence 17, 301–309.

Kennedy, J., Eberhart, R., Particle swarm optimization. In: Proceedings of the IEEE International Conference on Neural Networks, vol. 4, December 1995, pp. 1942–1948.

Kristinsson, K., Dumont, G.A., 1992. System identification and control using genetic algorithms. IEEE Transactions on Systems, Man and Cybernetics 22, 1033–1046.

Krusienski, D.J., Jenkins, W.K., 2005. Design and performance of adaptive systems based on structured stochastic optimization strategies. IEEE Circuits and Systems Magazine 5, 8–20.

Krusienski, D.J., Jenkins, W.K., 2004. Particle swarm optimization for adaptive IIR filter structures. In: Congress on Evolutionary Computation vol. 1, 965–970.

Kundur, P. Power System Stability and Control. McGraw-Hill, New York, 1993.

Lee, B.H., Kim, S., Seok, J., Won, S., Nonlinear system identification based on support vector machine using particle swarm optimization. In: International Joint Conference, SICE-ICASE, October 2006, pp. 5614–5618.

Liu J., Wenbo, X., Sun, J., Nonlinear system identification of Hammerstien and Wiener model using swarm intelligence. In: IEEE International Conference on Information Acquisition, August 2006, pp. 1219–1223.

Luitel B., Venayagamoorthy, G.K., A PSO with quantum infusion algorithm for training simultaneous recurrent neural networks. In: IEEE-INNS International Joint Conference on Neural Networks (IJCNN), June 2009, pp. 1923–1930.

Luitel B., Venayagamoorthy, G.K., Differential evolution Particle swarm optimization for digital filter design. In: Proceedings of the World Congress on Computational Intelligence, June 2008a, pp. 3954–3961.

Luitel B., Venayagamoorthy, G.K., Particle swarm optimization with quantum infusion for the design of digital filters. In: Proceedings of the Swarm Intelligence Symposium (SIS), September 2008b, pp. 1–8.

Mikki, S.M., Kishk, A., Quantum particle swarm optimization for electromagnetic. In: IEEE Transactions on Antennas and Propagation, vol. 54, October 2006.

Netto, S.L., Diniz, P.S.R., Agathoklis, P., 1995. Adaptive IIR filtering algorithms for system identification: a general framework. IEEE Transactions on Education 38, 54–66.

Ng, S.C., Leung, S.H., Chung, C.Y., Luk, A., Lau, W.H., 1996. The genetic search approach. A new learning algorithm for adaptive IIR filtering. IEEE Signal Processing Magazine, 38–46.

Panda, G. Mohanty, D., Majhi, B., Sahoo, G., Identification of nonlinear systems using particle swarm optimization technique. In: IEEE Conference on Evolutionary Computation, September 2007, pp. 3253–3257.

Shen, Y., Zeng C., A. Self-organizing particle swarm optimization algorithm and application. In: Third International Conference on Natural Computation, vol. 4, August 2007, pp. 668–672.

Shynk, J.J., 1989a. Adaptive IIR filtering. IEEE ASSP Magazine, 4–21.

Shynk, J.J., 1989b. Adaptive IIR filtering using parallel form realization. IEEE Transaction on Acoustic, Speech, Signal Processing 37 (4), 519–533.

Sun, J., Feng, B., Xu, W., 2004a. Particle swarm optimization with particles having quantum behavior. In: Congress on Evolutionary Computation 1, 325–331.

Sun, J., Xu, W., Feng, B., Adaptive parameter control for quantum-behaved particle swarm optimization on individual level. In: IEEE International Conference on Systems, Man and Cybernetics, vol. 4, October 2005, pp. 3049–3054.

Sun, J., Xu, W., Feng, B., Global search strategy of quantum-behaved particle swarm optimization. In: IEEE Conference on Cybernetics and Intelligent Systems, vol. 1, December 2004b, pp. 111–116.

Venayagamoorthy, G.K., 2007. Online design of an echo state network based wide area monitor for a multimachine power system. Neural Networks 20, 404–413.

Widrow, B., McCool, J.M., Larimore, M.G., Johnson, C.R. Jr., Stationary and nonstationary learning characteristics of the LMS adaptive filter. Proceedings of the IEEE 64 1976, pp. 1151–1162.

Please cite this article as: Luitel, B., Venayagamoorthy, G.K., Particle swarm optimization with quantum infusion for system identification. Engineering Applications of Artificial Intelligence (2010), doi:10.1016/j.engappai.2010.01.022