Soft Comput (2012) 16:1061–1069
DOI 10.1007/s00500-012-0803-y

ORIGINAL PAPER

An improved cooperative quantum-behaved particle swarm optimization

Yangyang Li · Rongrong Xiang · Licheng Jiao · Ruochen Liu

Published online: 24 January 2012

Abstract Particle swarm optimization (PSO) is a population-based stochastic optimization. Its parameters are easy to control, and it operates easily. But, the particle swarm optimization is a local convergence algorithm. Quantum-behaved particle swarm optimization (QPSO) overcomes this shortcoming, and outperforms original PSO. Based on classical QPSO, cooperative quantum-behaved particle swarm optimization (CQPSO) is present. This CQPSO, a particle firstly obtaining several individuals using Monte Carlo method and these individuals cooperate between them. In the experiments, five benchmark functions and six composition functions are used to test the performance of CQPSO. The results show that CQPSO performs much better than the other improved QPSO in terms of the quality of solution and computational cost.

Keywords Particle swarm optimization · Quantum-behaved · Cooperative quantum-behaved particle swarm optimization · Composition functions

1 Introduction

In recent years, many intelligent algorithms such as simulated annealing algorithm, Genetic algorithm, Taboo search and neural network have significantly grown in the last few years. Particle swarm optimization, a branch of evolutionary computation, was originally proposed by Kennedy and Eberhart (1995). PSO is a population-based random search technique and outperforms in many optimization problem particularly in nonlinear optimization problem. The PSO was inspired on the choreography of a bird flock.

PSO is simple and easy to understand. What's more, Parameters is less, easy programming and controlling. Since 2003, many improved swarm intelligence algorithms have been proposed, as in Zheng et al. (2003) and Clerc (2004). But, same as other intelligent methods, PSO has the disadvantage of prematurely. In many instances, it is easy to fall into local optimum. This is mainly because the trajectory is fixed, and the search space of every generation is limited. So, in limited iterations, many areas have no searches. Sun et al. (2004) introduced wave functions into PSO, proposing quantum-behaved particle swarm optimization (QPSO). He pointed out that the particles have quantum behavior and they move according to wave functions. Particles are cast off the bondage of trajectories. The global search ability has been greatly improved. After this, a series of studies of QPSO appear, such as the study for parameters in Sun et al. (2005a, b) and some applications of QPSO (Mikki and Kishk 2005; Fang et al. 2009; Sun et al. 2010a). Besides, several improvements of QPSO appear (Fang et al. 2010a) and theoretical research of QPSO algorithm has also been studied, such as (Fang et al. 2010b).

Especially in recent years, QPSO has been applied more and more widely (Coelho and Mariani 2008; Omkar 2009; Sabat et al. 2009). Leandro dos Santos Coelho (2008) proposed an improved quantum-behaved swarm optimization with chaotic mutation operator. Soon after, in 2010, more improved quantum-behaved swarm optimization was proposed, e.g., (Sun et al. 2010b; Lu et al. 2010).

With the problem more complex and dimension higher, getting the desired results is more and more difficult. In

Y. Li (✉) · R. Xiang · L. Jiao · R. Liu
Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education of China, Xidian University, Xi'an 710071, China
e-mail: yyli@xidian.edu.cn

order to take full of the uncertainty of quantum mechanics, a number of measurements are used when measuring the specific location. On this base, inspired by Gao et al. (2010), cooperative quantum-behaved particle swarm optimization (CQPSO) is proposed. In CQPSO, the information of particles having quantum behavior and information of every dimension of every particle are used. The details will be introduced in Sect. 3.

2 The quantum-behaved particle swarm optimization

In the classical PSO model, the ith particle is presented by X_i and V_i. The following equation is the evolutionary equation:

$$V_i(t+1) = V_i(t) + c1 * r1(t) * (P_i(t) - X_i(t)) + c2 * r2(t) * \left(P_g(t) - X_i(t)\right). \quad (1)$$

Shi (1998) proposed the widely used Standard PSO, and the parameter ω is added in the equation:

$$V_i(t+1) = \omega * V_i(t) + c1 * r1(t) * (P_i(t) - X_i(t)) + c2 * r2(t) * \left(P_g(t) - X_i(t)\right), \quad (2)$$

where X_i is the position of particles and V_i is the velocity of particles. $c1$ and $c2$ are the accelerated coefficients or learning factors, commonly $c1 = c2 = 2$ and usually $c1 = c2 \in (0, 4)$. $r1$ and $r2$ are uniformly distributed random number from 0 to 1. P_i is the personal best particle and P_g is the global best particle.

Then, a lot of improved PSO algorithms emerged. PSO have only three parameters: ω, $c1$ and $c2$. In the PSO, the concept is easy to understand and the parameters are very easy to control, easy programming and simple computing. In terms of memory and computing speed, the cost is low. But, the search depends on speed in PSO and the speed is limited, what's more, the particles only move along a fixed track, which results PSO has poor global ability, easy to converge to the local optimal solution. In another word, PSO is not a global optimization algorithm. In view of the shortcoming caused by certainty, take uncertainties into account. Then, the QPSO is proposed in Sun et al. (2004a).

QPSO employs a probability searching technique, and it is an uncertainty searching algorithm. In QPSO, researchers transfer the search space from classical space to quantum space, and the movement of the particles is the same as which of the ones with the quantum mechanics (Marchildon 2009).

The QPSO uses the wave principle of the particles in the microscopic world. In the iteration process, firstly initialize the position of the population, and then the particles search according to the wave function. In the QPSO, the particles could move in the search space. The particle appears anywhere in the searching space with a certain probability. To evaluate the individual, we need to learn the precise position of every particle and then obtain the fitness of the particles. In the quantum mechanics, the position is measured by Monte Carlo methods. At last, in QPSO, the particles update according to the following equations:

$$mbest = \left(\frac{1}{M}\sum_{i=1}^{M} P_{i1}, \frac{1}{M}\sum_{i=1}^{M} P_{i2}, \ldots, \frac{1}{M}\sum_{i=1}^{M} P_{id}\right), \quad (3)$$

$$P_i(t+1) = \varphi * P_i(t) + (1-\varphi) * P_g, \quad (4)$$

$$X_i(t+1) = p_i(t+1) \pm \alpha * |mbest - X_i(t)| * \ln(1/u), \quad (5)$$

where $mbest$ is the mean position and its expression is above. $P_i(t+1)$ contains the personal best position and the global best position. M is population size. φ and u are two random numbers uniformly distributed from 0 to 1. α is called Creativity Coefficient, the only parameter.

In Sabat et al. (2009), an improved QPSO algorithm with weighted mean best position (WQPSO) is proposed. In Xi et al. (2008), the author points out that the fitness of the individual is different, and then the contribution is also different for $mbest$. The nearer the best solution, the larger its contribution and weight coefficient are. The $mbest$, therefore, is:

$$mbest = \left(\frac{1}{M}\sum_{i=1}^{M} a_{i1}P_{i1}, \frac{1}{M}\sum_{i=1}^{M} a_{i2}P_{i2}, \ldots, \frac{1}{M}\sum_{i=1}^{M} a_{id}P_{id}\right), \quad (6)$$

where M is the population size and a_i is the weight coefficient and a_{id} is the dimension coefficient of every particle.

Currently, there has been many improved OPSO algorithms (Huang et al. 2009; Ankit Pat et al. 2010; Yang et al. 2010).

3 The cooperative quantum-behaved particle swarm optimization

QPSO is an uncertain and global random algorithm. In the search process, according to wave function, the particle could move anywhere in the searching space. To get the fitness, the precise position must be determined. This process is known as measurement and generally it is measured by the Monte Carlo method.

For high-dimensional functions, most algorithms are difficult to find global optimum. To get the relatively satisfactory solution, only increase the maximum iterations. As high dimension, computational cost increases. The traditional PSOs and QPSOs only retain the personal best and the global best. In Sun et al. (2004b), Hao Gao et al. proposed a thought of cooperation.

Inspired by the ideas, we have the following thought. The previous methods are only to get a position by measuring, and the other information it carries is lost out. Now, in order to take full advantage of quantum uncertainty, we obtain multiple solutions through multiple measurements and deal with these solutions by the cooperating operation. Figures 1 and 2 show the details.

From Fig. 1, we give the procedure of the CQPSO algorithm. The procedure is followed.

Procedure:

Step 1: Initialize the positions of the population of particles $p(t)$, $t = 1$.
Step 2: Calculate the fitness of the particles and generate the personal best individual $pbest(t)$ and the global best individual $gbest(t)$.
Step 3: Every particle generates five positions $x\_measure$ by multi-measurement according to Eqs. (3), (4) and (5) and then select the best position as the $X(t+1)$.
Step 4: Cooperate according to Fig. 2 and obtain $pbest(t+1)$ and $gbest(t+1)$.

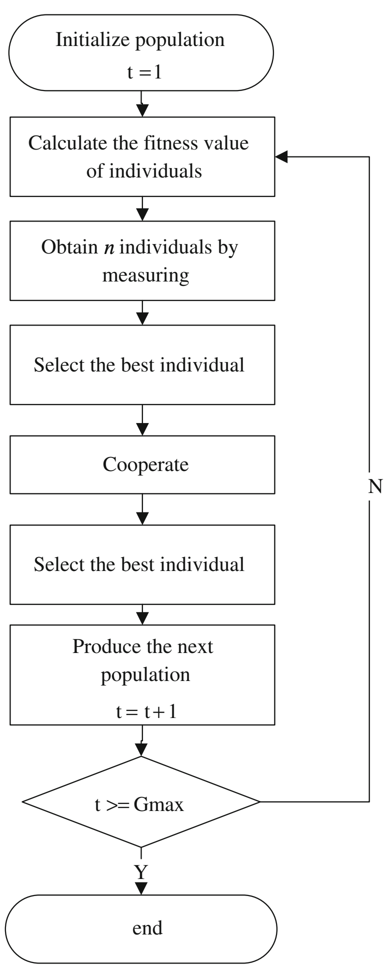

Fig. 1 Algorithm flow chart

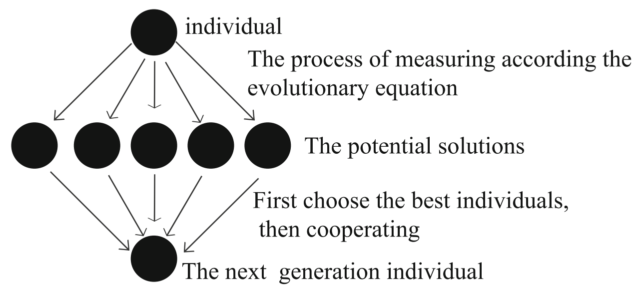

Fig. 2 The process of measurement and cooperation

Step 5: Go to Step 3 until the maximum iterations are met.

In this paper, the number of measurement is five, and the other parameters are the same as that in Xi et al. (2008). The detail is following. We initialize 20 individuals in the search space, and then every particle produces five particles through five measurements, $Measure\_num = 5$. $pbest_i$, $i = 1, 2, \ldots, 5$ is obtained. In another word, for any individual, give five u. According to expression (3), (4) and (5), we obtain five positions: $pbest_1$, $pbest_2$, $pbest_3$, $pbest_4$ and $pbest_5$. Now, if we indirectly keep back the best individual $pbest_1$ and abandon the other four individuals, we will lose much useful information. Inspired by Gao et al. (2010), in this step, add cooperation. As following, firstly, choose the best particle $pbest_1$ according to their fitness value from the five particles. Secondly, replace very dimension of the best particle by every dimension of the other particles. If the replaced particle's value is better, substitute the $pbest$. For example, ($pbest_1$) produces five positions ($pbest_{11}$, $pbest_{12}$, $pbest_{13}$, $pbest_{14}$, $pbest_{15}$), and suppose ($pbest_{11}$) is the best particle, $pbest_1 = pbest_{11}$. By this, replace x_{11} by x_{21}, and compute the value of the new particle $pbest'_{11} = (x_{21}, x_{12}, x_{13}, x_{14}, x_{15}, \ldots, x_{1n})$. If the new particle is better, $pbest'_{11} = pbest_1$. Next, for every dimension of every other particle, we do the same work. Figure 3 displays the process.

In our improved method, the number of measurements is a new parameter. To illustrate the influence of different times of measurements for convergence process, we do the following test. In the Figs. 4 and 5, from top to bottom are the convergence curves for $measurement = 1$, $measurement = 2$, $measurement = 3$, $measurement = 4$,

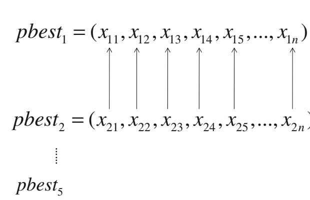

Fig. 3 Process of cooperating

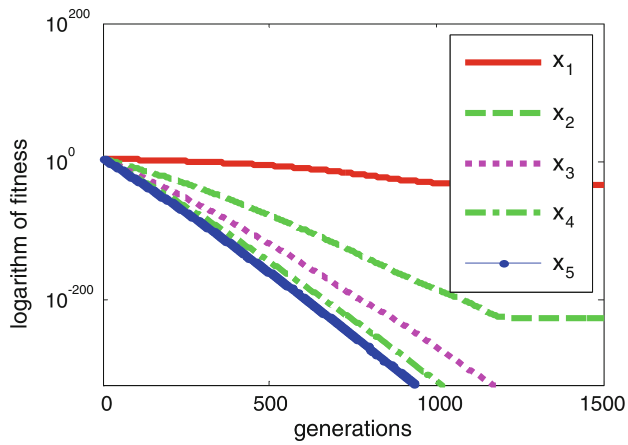

Fig. 4 Comparison of convergence process different times of measurements for f1

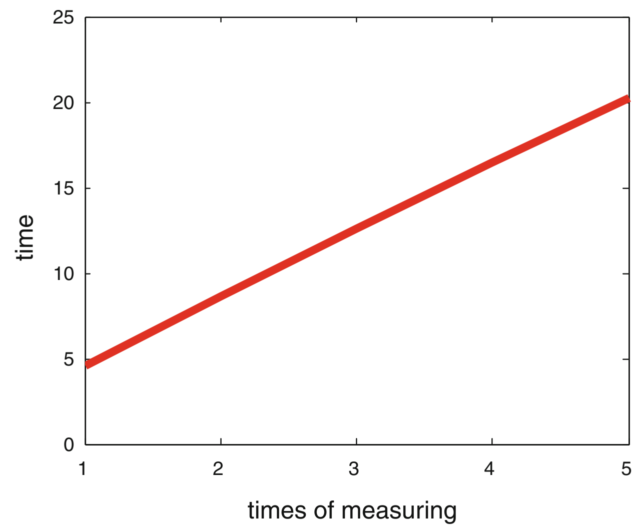

Fig. 6 The time of different numbers of measurements

measurement = 5. Obviously, the more the number of measurements, the farther the convergence speed is. At the same time, however, it is based on the cost of time in Fig. 6. Time is linearly growing with the growth of measurements. In our practical applications, we can change the times according to our need. In this paper, the number of the measurements is 5.

4 Experiment results and discussion

4.1 The experiments of Benchmark functions

To compare the performance of the three algorithms, experiment results of five benchmark functions will firstly display as follows. The functions and initial range of the population are listed in Table 1, asymmetry as used in (Sun et al. 2004b). These functions are all minimization problems with minimum value zeros.

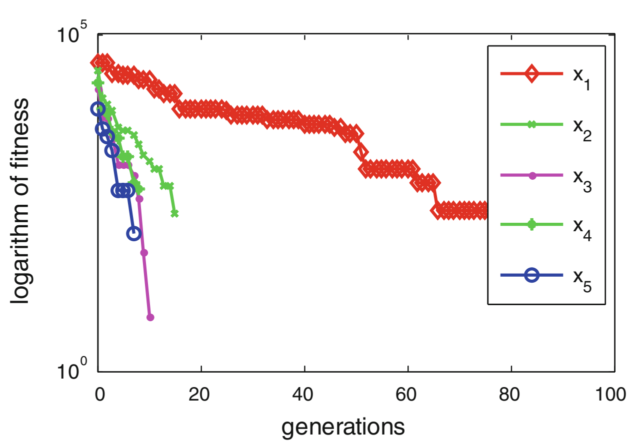

Fig. 5 Comparison of convergence process different times of measurements for F4

In the experiments, population is all 20. We empirically set the parameters as following. In QPSO, α decreases linearly from 1.0 to 0.5. In WQPSO, α also decreases linearly from 1.0 to 0.5, and the weight coefficient decreases from 1.5 to 0.5. In CQPSO, set *measurements* = 5, and the others are same as QPSO. We had 50 trial runs for every instance and recorded mean best fitness and standard deviation. In Sun et al. (2005), the dimensions are set as 10, 20, and 30. CQPSO works better for higher dimensions theoretically. Therefore, the population size is 20 and the maximum generation is 1,500, 2,000 and 3,000. The numerical value results are displayed in Table 2 and the performance of the three algorithms is shown in Figs. 4, 5, 6, 7, 8 and 9.

The numerical results in Table 2 show that the proposed CQPSO has better performance than QPSO and WQPSO for Sphere function, Rosenbrock function, Rastrigrin function and De Jong's function. For Griewank function, when dimension is 20, the result of CQPSO is no better than that of QPSO and WQPSO. However, along with the growing dimensions, the CQPSO obtains better result. This indicates the CQPSO outperforms in most applications, particularly in higher optimization problem.

Figures 7, 8, 9, 10 and 11 give the comparison results of the mean minimum and the distribution syllabify. The trial run number is 50, dimension is 20 and maximum generation is 1,500. For Sphere function, Rosenbrock function and De Jong's function, CQPSO show better performance. For Griewank function, although CQPSO has a little worse result, it has a better stability. For Rastrigrin function, CQPSO performs better than QPSO and WQPSO. In higher

Table 1 Basic characteristic of test functions

The functions	Expression	Initialization interval	Maximal area
Sphere function	$f1(x) = \sum_{i=1}^{n} x_i^2$	(−50, 100)	100
Rosenbrock function	$f2(x) = \sum_{i=1}^{n} \left(100\left(x_{i+1} - x_i^2\right) + (x_i - 1)^2\right)$	(15, 300)	100
Rastrigrin function	$f3(x) = \sum_{i=1}^{n} \left(x_i^2 - 10\cos(2\pi x_i) + 10\right)$	(2.56, 5.12)	10
Griewank function	$f4(x) = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	(−300, 600)	600
De Jong's function	$f5(x) = \sum_{i=1}^{n} ix_i^4$	(−30, 100)	100

Table 2 Comparison among QPSO and WQPSO

f	M	D	G_{max}	QPSO		WQPSO	
				Mean min	St. var	Mean min	St. var
f1	20	20	1,500	1.3208E−23	3.3218E−23	2.4267E−38	5.8824E−38
		30	2,000	1.5767E−15	4.0566E−15	6.9402E−32	1.2879E−31
		100	3,000	1.5077E+01	1.5926E+01	4.2014E−11	4.0006E−11
f2	20	20	1,500	9.0260E+01	1.3274E+02	4.4948E+01	5.8837E+01
		30	1,000	1.8484E+02	2.6972E+02	7.6625E+01	1.0193E+02
		100	3,000	1.9117E+05	1.7949E+05	2.4832E+02	1.9868E+02
f3	20	20	1,500	1.5697E+01	5.6303E+00	1.2945E+01	4.0725E+00
		30	2,000	3.0375E+01	7.3978E+00	2.4259E+01	7.9174E+00
		100	3,000	3.0458E+02	3.8518E+01	2.1121E+02	3.5535E+01
f4	20	20	1,500	1.8823E−02	1.9380E−02	2.4863E−02	2.3981E−02
		30	1,000	4.7967E−03	7.8878E−03	9.0994E−03	1.2641E−02
		100	3,000	1.1076E+00	3.3209E−01	4.4359E−03	9.0706E−03
f5	20	20	1,500	1.4444E−30	6.7034E−30	2.4224E−50	1.5425E−49
		30	2,000	3.0627E−18	1.1664E−17	1.5686E−40	5.9721E−40
		100	3,000	1.8609E+05	3.4648E+05	7.7518E−11	7.8925E−11

functions, the local search ability and global search ability are enhanced by the CQPSO.

4.2 The experiments of functions for the CEC2005 special session on real-parameter optimization

In order to validate the QPSO works better further, we test six functions, Shifted Schwefel's Problem 1.2 with Noise in Fitness (F4), Schwefel's Problem 2.6 with Global Optimum on Bounds (F5), Shifted Rotated Griewank's Function without Bounds (F7), Shifted Rotated Ackley's Function with Global Optimum on Bounds (F8), Expanded Extended Griewank's plus Rosenbrock's Function (F8, F2) (F13) and Shifted Rotated Expanded Scaffer's F6 (F14) in Lu et al. (2010). The results will be also compared with the QPSO and WQPSO. The function 4 and function 5 are mono-modal functions. The function 7 and function 8 are multimodal functions and basic functions. The function 13 and function 14 are multimodal functions and expanded functions. The concrete expressions and other details of functions are in Suganthan et al. (2005). The numerical

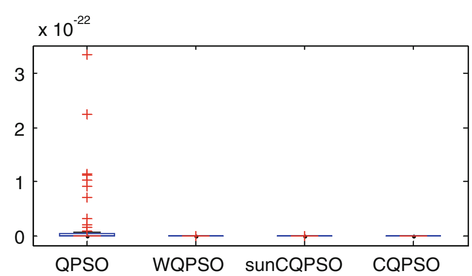

Fig. 7 Box figure of Sphere function

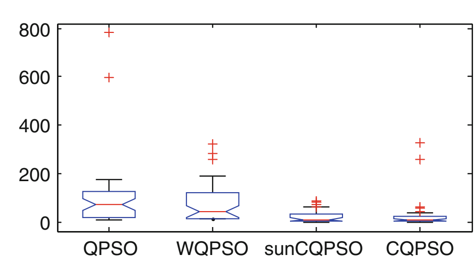

Fig. 8 Box figure of Rosenbrock function

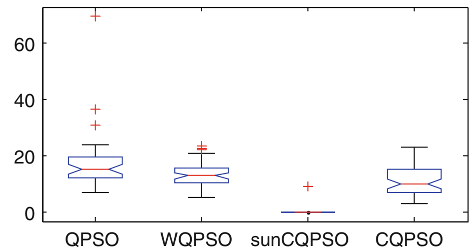

Fig. 9 Box figure of Rastrigrin function

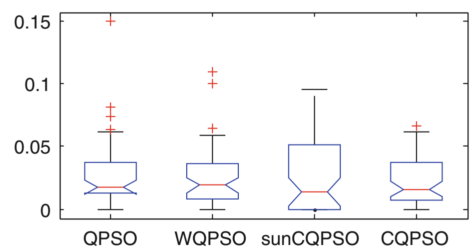

Fig. 10 Box figure of Griewank function

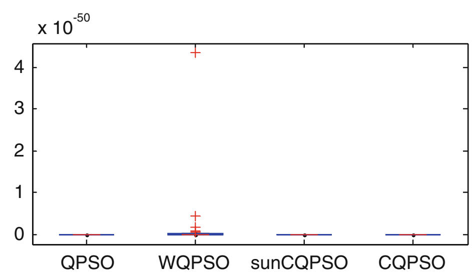

Fig. 11 Box figure of De Jong's function

value results are displayed in Tables 3 and 4. The comparison of performance is shown in Figs. 10, 11, 12, 13, 14, 15, 16 and 17.

Except Fig. 14 or function 7, in other figures, the conclusion is the proposed CQPSO generates the best results. In Table 5 and in Fig. 15, although the CQPSO obtained best results, we could see the CQPSO is all fell into local minimum. From the functional image, it can be seen that the global optimal solution is located in a narrow area and beside it, there is a local optimization. Because of the improved algorithm CQPSO still having large uncertainty, it is easy to jump out of the local optimization and at the same time it also jumps away from the area nearby the global optimal minimum. In other words, it tends to be far away from global optimization.

To further see the advantages of CQPSO, t test is done. The trial run number is 50 for Benchmark functions and is 25 for the CEC2005 special session on real-parameter optimization. The dimension is 20 for Benchmark functions and is 10 for the CEC2005 special session on real-parameter optimization. The maximum generation is 1,500 for Benchmark functions and is 1,000 for the CEC2005 special session on real-parameter optimization.

The results of comparing algorithms by one-tailed test with 98 degrees of freedom at a 0.05 level of significance are given in Table 6. The meaning of the symbol in the table is the same as (Yang and Yao 2008). From Table 6, we can get the same conclusion. CQPSO has better performance in the most instances.

From all the results above in the tables and figures, it can be concluded that the improved cooperative QPSO has

Table 3 Comparison among sunCQPSO and CQPSO

f	M	D	G_{max}	sunCQPSO		CQPSO	
				Mean min	St. var	Mean min	St. var
f1	20	20	1,500	4.946880E−317	0.0000E+00	**0.0000E+00**	**0.0000E+00**
		30	2,000	0.0000E+00	0.0000E+00	4.4466E−323	0.0000E+00
		100	3,000	2.4209E−218	0.0000E+00	3.6129E−98	2.5448E−97
f2	20	20	1,500	3.7499E+01	4.8401E+01	**2.9140E+01**	**5.6023E+01**
		30	1,000	5.5191E+01	6.4979E+01	**2.9660E+01**	**4.6534E+01**
		100	3,000	**8.4586E+01**	**4.2951E+01**	1.4697E+02	9.3562E+01
f3	20	20	1,500	**0.0000E+00**	**0.0000E+00**	1.2198E+01	6.4537E+00
		30	2,000	**5.9698E−02**	**2.3869E−01**	1.8049E+01	6.3279E+00
		100	3,000	**9.1352E+00**	**7.0400E+00**	1.1293E+02	1.7379E+01
f4	20	20	1,500	4.2273E−02	4.3296E−02	**1.9176E−02**	**1.6191E−02**
		30	1,000	6.1817E−02	6.9100E−02	**1.0279E−02**	**1.5108E−02**
		100	3,000	**4.4409E−18**	**2.1977E−17**	2.6110E−03	5.1311E−03
f5	20	20	1,500	0.0000E+00	0.0000E+00	**0.0000E+00**	**0.0000E+00**
		30	2,000	0.0000E+00	0.0000E+00	**0.0000E+00**	**0.0000E+00**
		100	3,000	**5.3694E−285**	**0.0000E+00**	3.7938E−104	1.4349E−103

Bold values represent the best results

Table 4 Mean function value and the standard deviations of QPSO and WQPSO

F	Min	D	G_{max}	QPSO		WQPSO	
				Mean min	St. var	Mean min	St. var
F4	−450	10	1,000	−449.9364	1.2208E−01	−448.0866	9.3343E−01
		30	3,000	3816.7280	3.4424E+03	2712.2660	2.0903E+03
		50	5,000	35606.9000	1.1361E+04	23635.2300	7.2601E+03
F5	−310	10	1,000	−309.9324	2.2213E−01	−305.4161	1.3473E+00
		30	3,000	3193.7680	1.0230E+03	3041.3850	1.0065E+03
		50	5,000	6630.8480	1.4831E+03	5908.6570	1.6307E+03
F7	−180	10	1,000	−179.5421	3.1670E−01	−179.0744	1.2846E−01
		30	3,000	−179.9663	2.8176E−02	−177.9586	3.1768E−01
		50	5,000	−179.8597	1.9968E−01	−176.1282	5.3809E−01
F8	−140	10	1,000	−119.5401	9.4892E−02	−119.5451	8.9466E−02
		30	3,000	−118.9841	5.2917E−02	−118.9753	5.6180E−02
		50	5,000	−118.8186	3.4865E−02	−118.8049	3.4791E−02
F13	−130	10	1,000	−128.7693	5.2452E−01	−128.5778	5.3839E−01
		30	3,000	−125.4201	2.3376E+00	−121.1874	2.3874E+00
		50	5,000	−117.7473	4.2147E+00	−108.7592	4.5567E+00
F14	120	10	1,000	−299.9426	3.8619E−02	−299.9370	3.5529E−02
		30	3,000	−299.7405	7.6293E−02	−299.7303	5.5841E−02
		50	5,000	−299.5802	1.4488E−01	−299.5641	1.1844E−01

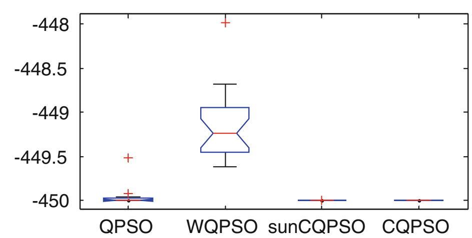

Fig. 12 Box figure of F4

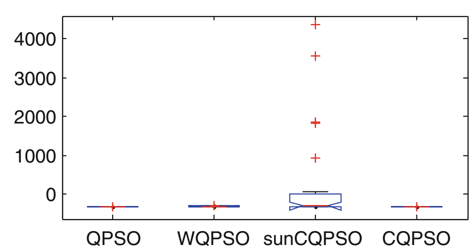

Fig. 13 Box figure of F5

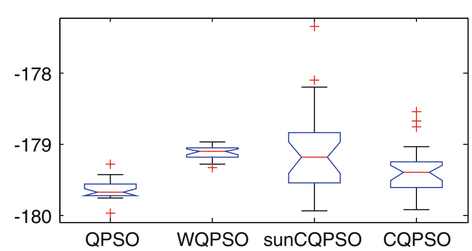

Fig. 14 Box figure of F7

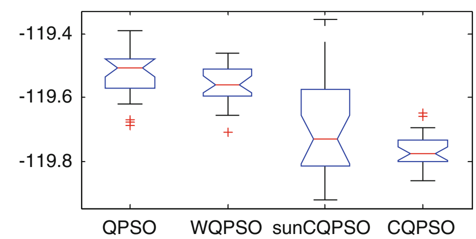

Fig. 15 Box figure of F8

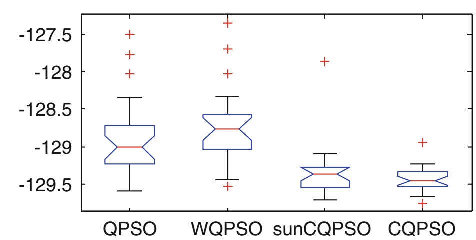

Fig. 16 Box figure of F13

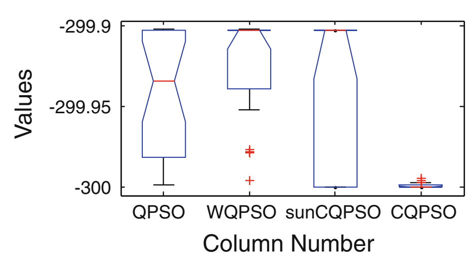

Fig. 17 Box figure of F14

Table 5 Mean function value and the standard deviations of sunCQPSO and CQPSO

F	Min	D	$G_{\max}$	sunCQPSO		CQPSO	
				Mean min	St.var	Mean min	St.var
F4	−450	10	1,000	−450.0000	2.5043E−06	**−450.0000**	**7.6966E−14**
		30	3,000	−358.6935	6.5472E+01	**−450.0000**	**1.7589E−10**
		50	5,000	10018.2800	5.3364E+03	**−449.9956**	**5.4487E−03**
F5	−310	10	1,000	294.0756	1.2651E+03	**−309.9992**	**3.2831E−03**
		30	3,000	7562.1470	2.0838E+03	**2628.1310**	**6.5155E+02**
		50	5,000	17936.7500	3.0209E+03	**4317.4460**	**1.1770E+03**
F7	−180	10	1,000	−179.0861	6.0709E−01	−179.2787	3.3788E−01
		30	3,000	**−179.9779**	**2.6466E−02**	−179.9778	1.7880E−02
		50	5,000	−178.1728	9.0499E+00	**−179.9955**	**9.1090E−03**
F8	−140	10	1,000	−119.6891	1.7260E−01	**−119.7683**	**4.4329E−02**
		30	3,000	**−119.4285**	**1.6130E−01**	−119.1764	4.0630E−02
		50	5,000	**−119.4007**	**2.0131E−01**	−118.9729	2.4966E−02
F13	−130	10	1,000	−129.3349	3.4762E−01	**−129.4101**	**1.6906E−01**
		30	3,000	**−128.2843**	**4.8307E−01**	−127.2798	3.5150E−01
		50	5,000	**−125.7872**	**1.6563E+00**	−125.5263	7.3139E−01
F14	120	10	1,000	−299.9339	4.6257E−02	**−299.9967**	**8.8522E−03**
		30	3,000	−299.7785	1.2705E−01	**−299.9952**	**5.0671E−03**
		50	5,000	−299.3326	1.7327E+00	**−299.9963**	**3.5155E−03**

Bold values represent the best results

Table 6 The *t* test results of comparing CQPSO, QPSO, WQPSO and sunCQPSO

t test result	f1	f2	f3	f4	f5	F4	F5	F7	F8	F13	F14
CQPSO-QPSO	s+	s+	s−	+	s+	+	+	s-	s+	s+	s+
CQPSO-WQPSO	+	s+	−	+	+	s+	s+	s+	s+	s+	s+
CQPSO-sunCQPSO	s+	+	s−	+	s+	+	s+	s+	s+	+	s+

better search ability than the other QPSOs and with the increase of dimension.

5 Conclusion

In this paper, the thought of cooperation was introduced into QPSO and the resultant CQPSO was proposed. From the results, it's obviously to be seen that CQPSO has better performance, particularly in higher functions. In other word, the cooperation solved the curse of dimensionality. However, the proposed algorithm has some drawbacks. As other swarm intelligence algorithm, the shortcoming of premature convergence has not been improved. In our future work, some prior knowledge could be introduced as the cooperative learning to improve prematurity in CQPSO.

What's more, the number of measurements used in this paper is five which is our empirical value. This is worth being on the search of the number of measurements. Through measuring, we could get multiple individuals and cooperate in this paper. In the future work, learning operator can be used to deal with these multiple individuals according to different application such as clustering problems and image segmentation.

Acknowledgments This work was supported by the National Natural Science Foundation of China (Nos. 61001202 and 60803098), the Provincial Natural Science Foundation of Shaanxi of China (Nos. 2009JQ8015, 2010JM8030 and 2010JQ8023), the China Postdoctoral Science Foundation Funded Project (Nos. 20080431228, 20090461283 and 20090451369), the China Postdoctoral Science Foundation Special Funded Project (No. 200801426), the Fundamental Research Funds for the Central Universities (Nos. JY10000902040, JY10000902039, JY10000903007 and K50510020011), and the Fund for Foreign Scholars in University Research and Teaching Programs (the 111 Project) (No. B07048).

References

Clerc M (2004) Discrete particle swarm optimization. In: New optimization techniques in engineering. Springer, Berlin

Coelho LDS, Mariani VC (2008) Particle swarm approach based on quantum mechanics and harmonic oscillator potential well for economic load dispatch with valve-point effects. Energy Convers Manag 49(11):3080–3085

dos Santos Coelho L (2008) A quantum particle swarm optimizer with chaotic mutation operator. Chaos, Solitons Fractals 37(5):1409–1418

Fang W, Sun J, Xu WB (2009) A new mutated quantum-behaved particle swarm optimizer for digital IIR filter design. EURASIP J Adv Signal Process 1–7

Fang W, Sun J, Ding YR, Wu XJ et al (2010a) A review of quantum-behaved particle swarm optimization. IETE Tech Rev 27(4):336–348

Fang W, Sun J, Xu WB (2010b) Convergence analysis of quantum-behaved particle swarm optimization algorithm and study on its control parameter. Acta Physica Sinica 59(6):28–36

Gao H, Xu WB, Sun J et al (2010) Multilevel thresholding for image segmentation through an improved quantum-behaved particle swarm algorithm. IEEE Trans Instrum Meas 59:934–946

Huang Z, Wang YJ, Yang CJ (2009) A new improved quantum-behaved particle swarm optimization model. In: IEEE conference on industrial electronics and applications, Xi'an, May 25–27, pp 1560–1564

Kennedy J, Eberhart R (1995) Particle swarm optimization. In: Proceedings of IEEE international conference on neural network, pp 1942–1948

Lu SF, Sun CF, Lu ZD (2010) An improved quantum-behaved particle swarm optimization method for short-term combined economic emission hydrothermal scheduling. Energy Convers Manag 51(3):561–571

Marchildon L (2009) Does quantum mechanics need interpretation? In: International conference on quantum, nano and micro technologies. Los Alamitos, California, February 01–07, pp 11–16

Mikki SM, Kishk AA (2005) Investigation of the quantum particle swarm optimization technique for electromagnetic applications. In: IEEE antennas and propagation society international symposium, vol 2A, July 3–8, pp 45–48

Omkar SN, Khandelwal R, Ananth TVS et al (2009) Quantum behaved Particle Swarm Optimization (QPSO) for multi-objective design optimization of composite structures. Expert Syst Appl 36(8):11312–11322

Pat A, Hota AR (2010) An improved quantum-behaved particle swarm optimization using fitness-weighted preferential recombination. In: 2010 Second World Congress on Nature and Biologically Inspired Computing, in Kitakyushu, Fukuoka, Japan, Dec. 15–17, pp 709–714

Sabat SL, dos Santos Coelho L, Abraham A (2009) MESFET DC model parameter extraction using quantum particle swarm optimization. Microelectron Reliab 49(6):660–666

Shi YH, Eberhart RC (1998) A modified particle swarm optimizer. In: IEEE international conference on evolutionary computation, Anchorage, Alaska, May 4–9, pp 1945–1950

Suganthan PN, Hansen N, Liang JJ et al (2005) Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization, Technical Report, Nanyang Technological University, Singapore. http://www.ntu.edu.sg/home/EPNSugan

Sun J, Feng B, Xu WB (2004) Particle swarm optimization with particles having quantum behavior. In: IEEE proceeding of congress on evolutionary computation, pp 325–331

Sun J, Xu WB, Feng B (2004) A global search strategy of quantum-behaved particle swarm optimization. In: Cybernetics and intelligent systems proceedings of the 2004 IEEE conference, pp 111–116

Sun J, Xu WB, Liu J (2005) Parameter selection of quantum-behaved particle swarm optimization. In: International conference on advances in natural computation 2005, Lecture Notes in Computer Science, vol 3612, pp 543–552

Sun J, Xu WB, Feng B (2005) Adaptive parameter control for quantum-behaved particle swarm optimization on individual level. In: Proceedings of IEEE international conference on systems, man and cybernetics, Big Island, HI, USA, pp 3049–3054

Sun J, Fang W, Xu WB (2010a) A quantum-behaved particle swarm optimization with diversity-guided mutation for the design of two-dimensional IIR digital filters. IEEE Trans Circuits Syst II 56(2):141–145

Sun J, Wu XJ, Fang W, Ding YR et al (2010) Multiple sequence alignment using the Hidden Markov Model trained by an improved quantum-behaved particle swarm optimization, Information Sciences, In Press, Corrected Proof, Available online 18 November 2010

Xi ML, Sun J, Xu WB (2008) An improved quantum-behaved particle swarm optimization algorithm with weighted mean best position. Appl Math Comput 205(2):751–759

Yang J, Xie JH (2010) An improved quantum-behaved particle swarm optimization algorithm. In: 2010 2nd international Asia conference on informatics in control, automation and robotics, pp 159–162

Yang SX, Yao X (2008) Population-based incremental learning with associative memory for dynamic environments. IEEE Trans Evol Comput 12(5):542–561

Zheng YL, Ma LH, Zhang LY (2003) Empirical study of particle swarm optimizer with an increasing inertia weight. In: Proceedings of IEEE congress on evolutionary computation, Canbella, Australia, pp 221–226