

# Quantum-Behaved Particle Swarm Optimization with Diversity-Maintained

Hai-xia Long and Shu-lei Wu

**Abstract** Quantum-behaved particle swarm optimization (QPSO) algorithm is a global-convergence-guaranteed algorithm, which outperforms original PSO in search ability but has fewer parameters to control. But QPSO algorithm is to be easily trapped into local optima as a result of the rapid decline in diversity. So this paper describes diversity-maintained into QPSO (QPSO-DM) to enhance the diversity of particle swarm and then improve the search ability of QPSO. The experiment results on benchmark functions show that QPSO-DM has stronger global search ability than QPSO and standard PSO.

**Keywords** Diversity · Quantum-behaved particle swarm optimization · Diversity-maintained · Benchmark function

## 1 Introduction

Particle swarm optimization (PSO) is a kind of stochastic optimization algorithms proposed by Kennedy and Eberhart [1] that can be easily implemented and is computationally inexpensive. The core of PSO is based on an analogy of the social behavior of flocks of birds when they search for food. PSO has been proved to be an efficient approach for many continuous global optimization problems. However, as demonstrated by Van Den Bergh [2], PSO is not a global-convergence-guaranteed algorithm because the particle is restricted to a finite sampling space for each of the iterations. This restriction weakens the global search ability of the algorithm and may lead to premature convergence in many cases.

---

H. Long (✉) · S. Wu  
Department of Computer Science and Technology, Hainan Normal University,  
Haikou 571158, China  
e-mail: 64169486@qq.com

Several authors developed strategies to improve on PSO. Clerc [3] suggested a PSO variant in which the velocity to the best point found by the swarm is replaced by the velocity to the current best point of the swarm, although he does not test this variant. Clerc [4] and Zhang et al. [5] dynamically change the size of the swarm according to the performance of the algorithm. Eberhart and Shi [6], He et al. [7] adopted strategies based on dynamically modifying the value of the PSO parameter called *inertia weight*. Various other solutions have been proposed for preventing premature convergence: objective functions that change over time [8]; noisy evaluation of the function objective [9]; repulsion to keep particles away from the optimum [10]; dispersion between particles that are too close to one another [11]; reduction in the attraction of the swarm center to prevent the particles clustering too tightly in one region of the search space [12]; hybrids with other metaheuristic such as genetic algorithms [13]; ant colony optimization [14], etc. An up-to-date overview of the PSO is introduced in [15].

Recently, a new variant of PSO, called quantum-behaved particle swarm optimization (QPSO) [16, 17], which is inspired by quantum mechanics and particle swarm optimization model. QPSO has only the position vector without velocity, so it is simpler than standard particle swarm optimization algorithm. Furthermore, several benchmark test functions show that QPSO performs better than standard particle swarm optimization algorithm. Although the QPSO algorithm is a promising algorithm for the optimization problems, like other evolutionary algorithm, QPSO also confronts the problem of premature convergence and decreases the diversity in the latter period of the search. Therefore, a lot of revised QPSO algorithms have been proposed since the QPSO had emerged. In Sun et al. [18], the mechanism of probability distribution was proposed to make the swarm more efficient in global search. Simulated annealing is further adopted to effectively employ both the ability to jump out of the local minima in simulated annealing and the capability of searching the global optimum in QPSO algorithm [19]. Mutation operator with Gaussian probability distribution was introduced to enhance the performance of QPSO in Coelho [20]. Immune operator based on the immune memory and vaccination was introduced into QPSO to increase the convergent speed using the characteristic of the problem to guide the search process [21].

In this chapter, QPSO with diversity-maintained (QPSO-DM) is introduced. This strategy is to prevent the diversity of particle swarm declining in the search of later stage.

The rest of the chapter is organized as follows. In Sect. 2, the principle of the PSO is introduced. The concept of QPSO is presented in Sect. 3, and the QPSO with diversity-maintained is proposed in Sect. 4. Section 5 gives the numerical results on some benchmark functions and discussion. Some concluding remarks and future work are presented in the last section.

## 2 PSO Algorithm

In the original PSO with  $M$  individuals, each individual is treated as an infinitesimal particle in the  $D$ -dimensional space, with the position vector and velocity vector of particle  $i$ ,  $X_i(t) = (X_{i1}(t), X_{i2}(t), \dots, X_{iD}(t))$ , and  $V_i(t) = (V_{i1}(t), V_{i2}(t), \dots, V_{iD}(t))$ . The particle moves according to the following equations:

$$V_{ij}(t+1) = V_{ij}(t) + c_1 \cdot r_1 \cdot (P_{ij}(t) - X_{ij}(t)) + c_2 \cdot r_2 \cdot (P_{gj}(t) - X_{ij}(t)) \quad (1)$$

$$X_{ij}(t+1) = X_{ij}(t) + V_{ij}(t+1) \quad (2)$$

for  $i = 1, 2, \dots, M; j = 1, 2, \dots, D$ . The parameters  $c_1$  and  $c_2$  are called the acceleration coefficients. Vector  $P_i = (P_{i1}, P_{i2}, \dots, P_{iD})$  known as the *personal best position* is the best previous position (the position giving the best fitness value so far) of particle  $i$ ; vector  $P_g = (P_{g1}, P_{g2}, \dots, P_{gD})$  is the position of the best particle among all the particles and is known as the *global best position*. The parameters  $r_1$  and  $r_2$  are two random numbers distributed uniformly in  $(0, 1)$ , that is,  $r_1, r_2 \sim U(0, 1)$ . Generally, the value of  $V_{ij}$  is restricted in the interval  $[-V_{\max}, V_{\max}]$ .

Many revised versions of PSO algorithm are proposed to improve the performance since its origin in 1995. Two most important improvements are the version with an Inertia Weight [22] and a Constriction Factor [23]. In the inertia-weighted PSO, the velocity is updated by using

$$V_{ij}(t+1) = w \cdot V_{ij}(t) + c_1 \cdot r_1 \cdot (P_{ij}(t) - X_{ij}(t)) + c_2 \cdot r_2 \cdot (P_{gj} - X_{ij}(t)) \quad (3)$$

while in the Constriction Factor model, the velocity is calculated by using

$$V_{ij}(t+1) = K \cdot [V_{ij}(t) + c_1 \cdot r_1 \cdot (P_{ij}(t) - X_{ij}(t)) + c_2 \cdot r_2 \cdot (P_{gj} - X_{ij}(t))] \quad (4)$$

where

$$k = \frac{2}{\left| 2 - \varphi - \sqrt{\varphi^2 - 4\phi} \right|} \quad \varphi = c_1 + c_2, \quad \varphi > 4 \quad (5)$$

The inertia-weighted PSO was introduced by Shi and Eberhart [6] and is known as the standard PSO.

## 3 QPSO Algorithm

Trajectory analyses in Clerc and Kennedy [24] demonstrated the fact that convergence of PSO algorithm may be achieved if each particle converges to its local attractor  $p_i = (p_{i1}, p_{i2}, \dots, p_{iD})$  with coordinates

$$\begin{aligned} p_{ij}(t) &= (c_1 r_1 P_{ij}(t) + c_2 r_2 P_{gj}(t)) / (c_1 r_1 + c_2 r_2), \text{ or } p_{ij}(t) \\ &= \varphi \cdot P_{ij}(t) + (1 - \varphi) \cdot P_{gj}(t) \end{aligned} \quad (6)$$

where  $\varphi = c_1 r_1 / (c_1 r_1 + c_2 r_2)$ . It can be seen that the local attractor is a stochastic attractor of particle  $i$  that lies in a hyper-rectangle with  $P_i$  and  $P_g$  being two ends of its diagonal. We introduce the concepts of QPSO as follows.

Assume that each individual particle moves in the search space with a  $\delta$  potential on each dimension, of which the center is the point  $p_{ij}$ . For simplicity, we consider a particle in one-dimensional space, with point  $p$  the center of potential. Solving Schrödinger equation of one-dimensional  $\delta$  potential well, we can get the probability distribution function  $D(x) = e^{-2|p-x|/L}$ . Using Monte Carlo method, we obtain

$$x = p \pm \frac{L}{2} \ln(1/u), \quad u \sim U(0, 1) \quad (7)$$

The above is the fundamental iterative equation of QPSO.

In Sun et al. [17], a global point called Mainstream Thought or Mean Best Position of the population is introduced into PSO. The mean best position, denoted as  $C$ , is defined as the mean of the personal best positions among all particles. That is

$$\begin{aligned} C(t) &= (C_1(t), C_2(t), \dots, C_D(t)) \\ &= \left( \frac{1}{M} \sum_{i=1}^M P_{i1}(t), \frac{1}{M} \sum_{i=1}^M P_{i2}(t), \dots, \frac{1}{M} \sum_{i=1}^M P_{iD}(t) \right) \end{aligned} \quad (8)$$

where  $M$  is the population size and  $P_i$  is the personal best position of particle  $i$ . Then, the value of  $L$  is evaluated by  $L = 2\alpha \cdot |C_j(t) - X_{ij}(t)|$ , and the position is updated by

$$X_{ij}(t+1) = p_{ij}(t) \pm \alpha \cdot |C_j(t) - X_{ij}(t)| \cdot \ln(1/u) \quad (9)$$

where parameter  $\alpha$  is called Contraction–Expansion (CE) Coefficient, which can be tuned to control the convergence speed of the algorithms. Generally, we always call the PSO with Eq. (9) quantum-behaved particle swarm optimization (QPSO). In most cases,  $\alpha$  decrease linearly from can be controlled to  $\alpha_0$  to  $\alpha_1$  ( $\alpha_0 < \alpha_1$ ). We outline the procedure of the QPSO algorithm as follows:

Procedure of the QPSO algorithm:

- Step 1: Initialize the population;
- Step 2: Computer the personal position and global best position;
- Step 3: Computer the mean best position  $C$ ;
- Step 4: Properly select the value of  $\alpha$ ;
- Step 5: Update the particle position according to Eq. (9);
- Step 6: While the termination condition is not met, return to Step 2;
- Step 7: Output the results.

## 4 QPSO-DM Algorithm

QPSO is a promising optimization problem solver that outperforms PSO in many real application areas. First of all, the introduced exponential distribution of positions makes QPSO global convergent. In the QPSO algorithm in the initial stage of search, as the particle swarm initialization, its diversity is relatively high. In the subsequent search process, due to the gradual convergence of the particle, the diversity of the population continues to decline. As the result, the ability of local search ability is continuously enhanced, and the global convergence ability is continuously weakened. In early and middle search, reducing the diversity of particle swarm optimization for contraction efficiency improvement is necessary; however, in late stage of search, because the particles are gathered in a relatively small range, particle swarm diversity is very low, the global search ability becomes very weak, the ability for a large range of search has been very small, and the phenomenon of premature will occur in this algorithm.

To overcome this shortcoming, we introduce diversity-maintained into QPSO.

The population diversity of the QPSO-DM is denoted as  $\text{diversity}(pbest)$  and is measured by average Euclidean distance from the particle's personal best position to the mean best position, namely

$$\text{diversity}(pbest) = \frac{1}{M \cdot |A|} \sum_{i=1}^M \sqrt{\sum_{j=1}^D (pbest_{i,j} - p\bar{best}_j)^2} \quad (10)$$

where  $M$  is the population of the particle,  $|A|$  is the length of longest the diagonal in the search space, and  $D$  is the dimension of the problem. Hence, we may guide the search of the particles with the diversity measures when the algorithm is running.

In the QPSO-DC algorithm, only low-bound  $d_{low}$  is set for  $\text{diversity}(pbest)$  to prevent the diversity from constantly decreasing. The procedure of the algorithm is as follows. After initialization, the algorithm is running in convergence mode. In process of convergence, the convergence mode is realized by Contraction–Expansion (CE) Coefficient  $\alpha$ . On the course of evolution, if the diversity measure  $\text{diversity}(pbest)$  of the swarm drops to below the low-bound  $d_{low}$ , the particle swarm turns to be in explosion mode in which the particles are controlled to explode to increase the diversity until it is larger than  $d_{low}$ .

## 5 Experiment Results and Discussion

To test the performance of the QPSO with diversity-maintained (QPSO-DM), six widely known benchmark functions listed in Table 1 are tested for comparison with standard PSO (SPSO), QPSO. These functions are all minimization problems with minimum objective function values as zeros. The initial range of the

**Table 1** Expression of the five tested benchmark functions

	Function expression	Search domain
Sphere	$f_1(X) = \sum_{i=1}^n x_i^2$	$-100 \leq x_i \leq 100$
Rosenbrock	$f_2(X) = \sum_{i=1}^{n-1} (100 \cdot (x_{i+1} - x_i^2)^2 + (x_i - 1)^2)$	$-100 \leq x_i \leq 100$
Rastrigrin	$f_3(X) = \sum_{i=1}^n (x_i^2 - 10 \cdot \cos(2\pi x_i) + 10)$	$-10 \leq x_i \leq 10$
Greiwank	$f_4(X) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{ i }}\right) + 1$	$-600 \leq x_i \leq 600$
Ackley	$f_5 = 20 + e - 20e^{-\frac{1}{3}\sqrt{\frac{1}{n}\sum_{i=1}^n x_i^2}} - e^{-\frac{1}{n}\sum_{i=1}^n \cos(2\pi x_i)}$	$-30 \leq x_i \leq 30$
Schwefel	$f_6 = 418.9829n - \sum_{i=1}^n (x_i \sin \sqrt{ x_i })$	$-500 \leq x_i \leq 500$

**Table 2** The initial range of population for all the tested algorithms and  $V_{\max}$  for spso

	Initial range	$V_{\max}$
$f_1$	(50, 100)	100
$f_3$	(15, 30)	100
$f_3$	(2.56, 5.12)	10
$f_4$	(300, 600)	600
$f_5$	(15, 30)	30
$f_6$	(250, 500)	500

population listed in Table 2 is asymmetry as used in Shi and Eberhart [25]. Table 2 also lists  $V_{\max}$  for SPSO. The fitness value is set as function value, and the neighborhood of a particle is the whole population.

As in Angeline [22], for each function, three different dimension sizes are tested. They are dimension sizes 10, 20, and 30. The max number of generations is set as 1,000, 1,500, and 2,000 corresponding to the dimensions 10, 20, and 30 for first six functions, respectively. The maximum generation for the last function is 2,000. In order to investigate whether the QPSO-DM algorithm is good or not, different population sizes are used for each function with different dimension. They are population sizes 20, 40, and 80. For SPSO, the acceleration coefficients are set to be  $c_1 = c_2 = 2$ , and the inertia weight is decreasing linearly from 0.9 to 0.4 as in Shi and Eberhart [25]. In experiments for QPSO, the value of CE Coefficient varies from 1.0 to 0.5 linearly over the running of the algorithm as in [18], while in QPSO-DM, the value of CE Coefficient is listed in Table 3. From the Table 3, we also obtain the CE coefficient of QPSO-DM decreases from 0.8–0.5 linearly. We had 50 trial runs for every instance and recorded mean best fitness and standard deviation.

The mean values and standard deviations of best fitness values for 50 runs of each function are recorded in Tables 4, 5, 6, 7, and 8.

The results show that both QPSO and QPSO-DM are superior to SPSO except on Schwefel and Shaffer’s  $f_6$  function. On Sphere Function, the QPSO works better than QPSO-DM when the warm size is 40 and dimension is 10, and when the warm size is 80 and dimension is 20. Except for the above two instances, the best result is QPSO-DM. The Rosenbrock function is a monomodal function, but

**Table 3** Parameter value of QPSO-dM

CE coefficient	Sphere function		Rosenbrock function		Rastrigrin function	
	Mean best	St. dev.	Mean best	St. dev.	Mean best	St. dev.
(1.0, 0.5)	1.2237e-013	1.9534e-013	75.0328	52.2408	88.5367	14.3207
(0.9, 0.5)	2.4890e-017	2.6749e-017	54.3227	69.7425	90.7490	18.2660
(0.8, 0.5)	<b>3.2054e-026</b>	1.6573e-025	<b>33.2065</b>	28.2984	78.4024	11.3524
(0.7, 0.5)	9.7934e-022	4.2890e-021	76.4626	88.0403	77.6020	9.8336
(1.0, 0.4)	4.6827e-012	2.1674e-011	39.7421	36.8124	69.7656	15.9507
(0.9, 0.4)	1.9889e-014	6.3148e-014	71.6725	89.3760	92.8337	13.6210
(0.8, 0.4)	5.9680e-015	2.5249e-014	69.5267	97.4625	83.3520	9.8675
(0.7, 0.4)	1.5027e-018	4.2109e-018	86.2217	122.5409	73.4665	17.6453
(1.0, 0.3)	1.8963e-010	4.5227e-010	56.3226	68.2468	59.2417	18.6628
(0.9, 0.3)	2.0527e-011	5.3745e-011	70.6849	98.7489	47.5948	15.0963
(0.8, 0.3)	5.6849e-014	6.1342e-013	90.3573	92.5246	61.0499	16.4421
(0.7, 0.3)	2.5864e-013	8.3346e-013	121.7752	145.4575	<b>40.7232</b>	18.3726

CE coefficient	Griewank function		Ackley function		Schwefel function	
	Mean best	St. dev.	Mean best	St. dev.	Mean best	St. dev.
(1.0, 0.5)	0.0328	0.0307	3.0209e-013	4.3354e-013	4.9953e + 003	216.7453
(0.9, 0.5)	0.0243	0.0152	3.2649e-014	8.3201e-014	4.8774e + 003	206.1504
(0.8, 0.5)	<b>0.0065</b>	0.0064	<b>1.1453e-014</b>	8.4250e-015	<b>2.8249e + 003</b>	415.4217
(0.7, 0.5)	0.0187	0.0097	3.9826e-014	5.6935e-014	4.1370e + 003	336.8420
(1.0, 0.4)	0.0651	0.0086	3.7380e-011	1.3192e-010	4.9682e + 003	209.5413
(0.9, 0.4)	0.0281	0.0170	1.5916e-012	3.8720e-012	5.1393e + 003	278.3024
(0.8, 0.4)	0.0106	0.0107	8.0437e-013	1.2495e-012	4.9504e + 003	255.6286
(0.7, 0.4)	0.0164	0.0165	0.0213	0.1634	4.5336e + 003	492.6527
(1.0, 0.3)	0.0137	0.0136	4.0969e-010	7.6405e-010	4.9972e + 003	252.7829
(0.9, 0.3)	0.0190	0.0190	7.0642e-011	1.1408e-010	5.0918e + 003	213.0963
(0.8, 0.3)	0.0151	0.0182	1.3296e-010	2.2764e-010	4.0437e + 003	324.4462
(0.7, 0.3)	0.0137	0.0126	0.1862	0.1324	3.8596e + 003	264.1462

its optimal solution lies in a narrow area that the particles are always apt to escape. The experiment results on Rosenbrock function show that the QPSO-DM outperforms the QPSO. Rastrigrin function and Griewank function are both multimodal and usually tested for comparing the global search ability of the algorithm. On Rastrigrin function, it is also shown that the QPSO-DM generated best results than QPSO. On Griewank function, QPSO-DM has better performance than QPSO and PSO algorithms. On Ackley function, QPSO-DM has best performance when the dimension is 10; except for these, the QPSO-DM has minimal value. On the Schwefel function, SPSO has the best performance in any situation. QPSO-DM is better than QPSO. Generally speaking, the QPSO-DM has better global search ability than SPSO and QPSO.

Figure 1 shows the convergence process of the three algorithms on the first four benchmark functions with dimension 30 and swarm size 40 averaged on 50 trail runs. It is shown that, although QPSO-DM converges more slowly than the QPSO

**Table 4** Numerical results on sphere function

M	Dim.	Gmax	SPSO			QPSO			QPSO-DM		
			Mean best	St. Dev.	St. dev.	Mean best	St. dev.	St. dev.	Mean best	St. dev.	St. dev.
20	10	1000	4.6119e-021	1.0352e-020	3.1979e-043	2.2057e-042	2.2057e-042	4.9172e-045	1.4682e-045	1.4682e-045	
	20	1500	9.0266e-012	3.5473e-011	1.9197e-024	7.1551e-024	7.1551e-024	1.2247e-026	2.6428e-026	2.6428e-026	
	30	2000	3.9672e-008	7.0434e-008	7.3736e-015	2.1796e-014	2.1796e-014	2.4369e-016	4.0917e-016	4.0917e-016	
40	10	1000	1.3178e-024	3.7737e-024	2.7600e-076	1.8954e-075	1.8954e-075	7.8025e-074	1.6432e-074	1.6432e-074	
	20	1500	2.3057e-015	6.6821e-015	3.9152e-044	1.8438e-043	1.8438e-043	4.1583e-045	7.6524e-045	7.6524e-045	
	30	2000	1.1286e-010	2.2994e-010	2.6424e-031	6.3855e-031	6.3855e-031	2.3208e-032	1.0527e-032	1.0527e-032	
80	10	1000	1.6097e-028	7.1089e-028	2.5607e-103	6.4847e-103	6.4847e-103	2.1084e-104	1.6582e-104	1.6582e-104	
	20	1500	6.3876e-018	1.4821e-017	1.4113e-068	7.4451e-068	7.4451e-068	1.3586e-068	3.7647e-068	3.7647e-068	
	30	2000	3.2771e-013	7.5971e-013	6.5764e-050	3.6048e-049	3.6048e-049	1.9688e-050	1.7052e-050	1.7052e-050	



**Table 5** Numerical results on Rosenbrock function

M	Dim.	Gmax	SPSO		QPSO		QPSO-DM	
			Mean best	St. dev.	Mean best	St. dev.	Mean best	St. dev.
20	10	1,000	51.0633	153.7913	9.5657	16.6365	6.7435	1.6753
	20	1,500	100.2386	140.9822	82.4294	138.2429	30.2483	22.5409
	30	2,000	160.4400	214.0316	98.7948	122.5744	64.2516	48.3277
40	10	1,000	24.9641	49.5707	8.9983	17.8202	3.0418	2.4972
	20	1,500	59.8256	95.9586	40.7449	41.1751	12.4916	10.3762
	30	2,000	124.1786	269.7275	43.5582	38.0533	34.5349	21.8962
80	10	1,000	19.0259	41.6069	6.8312	0.3355	4.0263	3.2651
	20	1,500	40.2289	46.8491	33.5287	31.6415	14.8572	1.3806
	30	2,000	56.8773	57.8794	44.5946	31.6739	20.5704	1.6127

**Table 6** Numerical results on rastrigrin function

M	Dim.	Gmax	SPSO		QPSO		QPSO-DM	
			Mean best	St. dev.	Mean best	St. dev.	Mean best	St. dev.
20	10	1,000	5.8310	2.5023	4.0032	2.1409	1.8261	0.3362
	20	1,500	23.3922	6.9939	15.0648	6.0725	11.0704	10.5382
	30	2,000	51.1831	12.5231	28.3027	12.5612	20.4728	3.6157
40	10	1,000	3.7812	1.4767	2.6452	1.5397	0.9504	1.3854
	20	1,500	18.5002	5.5980	11.3109	3.5995	10.3207	6.4872
	30	2,000	39.5259	10.3430	18.9279	4.8342	13.4697	5.7639
80	10	1,000	2.3890	1.1020	2.2617	1.4811	0.9650	1.0348
	20	1,500	12.8594	3.6767	8.4121	2.5798	5.7236	3.7684
	30	2,000	30.2140	7.0279	14.8574	5.0408	11.6570	4.1366
<i>Numerical results on griewank function</i>								
20	10	1,000	0.0920	0.0469	0.0739	0.0559	0.0425	0.0652
	20	1,500	0.0288	0.0285	0.0190	0.0208	0.0163	0.0216
	30	2,000	0.0150	0.0145	0.0075	0.0114	0.0077	0.0516
40	10	1,000	0.0873	0.0430	0.0487	0.0241	0.0468	0.0724
	20	1,500	0.0353	0.0300	0.0206	0.0197	0.0122	0.0364
	30	2,000	0.0116	0.0186	0.0079	0.0092	0.0013	0.0108
80	10	1,000	0.0658	0.0266	0.0416	0.0323	0.0388	0.0528
	20	1,500	0.0304	0.0248	0.0137	0.0135	0.0041	0.0086
	30	2,000	0.0161	0.0174	0.0071	0.0109	1.0273e-006	4.9264e-006

during the early stage of search, it may catch up with QPSO at later stage and could generate better solutions at the end of search.

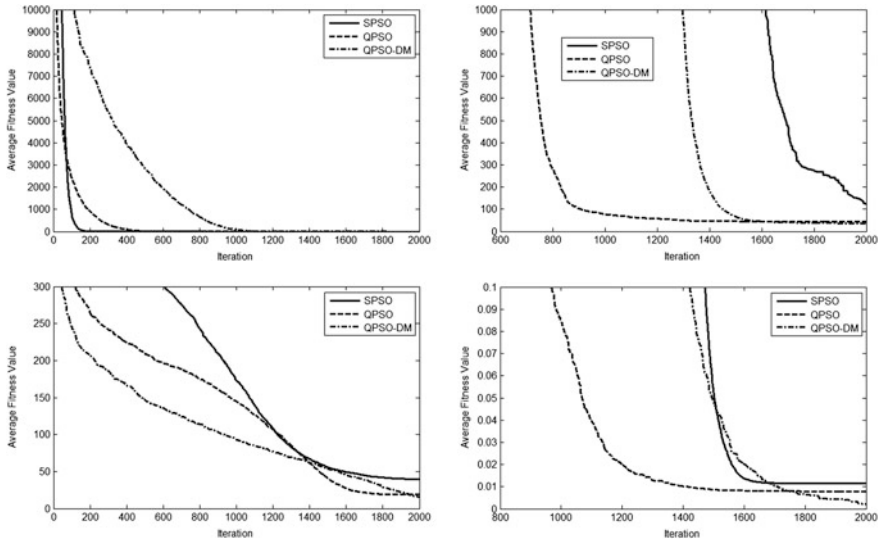
From the results above in the tables and figures, it can be concluded that the QPSO-DM has better global search ability than SPSO and QPSO.

**Table 7** Numerical results on ackley function

M	Dim.	Gmax	SPSO		QPSO		QPSO-DM	
			Mean best	St. dev.	Mean best	St. dev.	Mean best	St. dev.
			20	1000	2.0489e-011	3.0775e-011	6.8985e-012	1.2600e-011
20	1500	0.0285	0.2013	1.5270e-008	2.1060e-008	8.6631e-016	4.7815e-016	
30	2000	0.2044	0.4899	4.3113e-007	4.4188e-007	6.6520e-014	2.1354e-013	
40	1000	2.4460e-013	5.2901e-013	2.5935e-015	5.0243e-016	1.8209e-016	3.2296e-017	
20	1500	2.6078e-008	4.0653e-008	6.3491e-013	1.3305e-012	4.6917e-016	2.2475e-016	
30	2000	3.7506e-006	6.4828e-006	5.5577e-011	8.6059e-011	5.6051e-016	7.8722e-016	
80	1000	5.5778e-015	7.4138e-015	2.4514e-015	8.5229e-016	3.8246e-016	2.6115e-016	
20	1500	5.2979e-010	7.2683e-010	5.8620e-015	1.6446e-015	3.6504e-016	4.7535e-017	
30	2000	1.6353e-007	2.1300e-007	6.7253e-014	4.4759e-014	4.5725e-016	1.8864e-016	

**Table 8** Numerical results on schwefel function

M	Dim.	Gmax	SPSO			QPSO			QPSO-DM		
			Mean best	St. dev.	Mean best	St. dev.	Mean best	St. dev.	Mean best	St. dev.	
20	10	1,000	630.6798	169.9164	1.0784e + 003	342.1967	421.0562	188.3162			
	20	1,500	2.0827e + 003	327.0313	4.0026e + 003	679.1461	2.8447e + 003	1.2683e + 003			
	30	2,000	4.1097e + 003	481.8801	6.5075e + 003	589.9931	5.6023e + 003	1.4219e + 003			
40	10	1,000	494.0236	169.5594	1.4359e + 003	246.9290	628.2261	290.3428			
	20	1,500	1.8516e + 003	273.8690	4.5580e + 003	242.8220	2.1046e + 003	756.1462			
	30	2,000	3.4042e + 003	384.3150	7.3380e + 003	288.9463	6.5209e + 003	741.2416			
80	10	1,000	426.0796	130.5357	1.6060e + 003	132.0009	528.3466	273.1564			
	20	1,500	1.5466e + 003	270.5587	4.6407e + 003	225.9431	2.5592e + 003	425.3320			
	30	2,000	3.0325e + 003	419.8580	8.0583e + 003	226.3796	6.3488e + 003	560.3284			



**Fig. 1** Convergence process of the three algorithms on the first four benchmark functions with dimension 30 and swarm size 40 averaged on 50 trail runs

**Acknowledgments** This work is supported by National Natural Science Fund (No. 61163042), Higher School Scientific Research Project of Hainan Province (Hjkj2013-22), International Science and Technology Cooperation Program of China (2012DFA11270), and Hainan International Cooperation Key Project (GJXM201105).

## References

1. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proceedings IEEE International Conference Neural Networks, pp.1942–1948 (1995)
2. Van den Bergh, F.: An Analysis of Particle Swarm Optimizers. University of Pretoria, South Africa (2001)
3. Clerc, M.: Discrete particle swarm optimization illustrated by the traveling salesman problem. New optimization techniques in engineering, Berlin: Springer pp. 219–239 (2004)
4. Clerc, M.: Particle swarm optimization, ISTE, 2006
5. Zhang, W., Liu, Y., Clerc, M.: An adaptive PSO algorithm for reactive power optimization. In: 6th International Conference on Advances in Power Control, Operation and Management, Hong Kong (2003)
6. Eberhart, R.C., Shi, Y.: Comparing inertia weights and constriction factors in particle swarm optimization. In: IEEE International Conference on Evolutionary Computation, pp. 81–86 (2001)
7. He, S., Wu, Q.H., Wen, J.Y., Saunders, J.R., Paton, R.C.: A particle swarm optimizer with passive congregation. *Biosystems* **78**, 135–147 (2004)
8. Hu, X., Eberhart, R.C.: Tracking dynamic systems with PSO: where's the cheese?. In: Proceedings of the Workshop on Particle Swarm Optimization, Indianapolis (2001)
9. Parsopoulos, K.E., Vrahatis, M.N.: Particle swarm optimizer in noisy and continuously changing environments. *Artif. Intell. Soft Comput.*, 289–294 (2001)

10. Parsopoulos, K.E., Vrahatis, M.N.: On the computation of all global minimizers thorough particle swarm optimization. *IEEE Trans. Evol. Comput.* **8**, 211–224 (2004)
11. LoZvbjerg, M., Krink, T: Extending particle swarms with self-organized criticality. In: *Proceedings of the IEEE congress on evolutionary computation*, pp. 1588–1593 (2002)
12. Blackwell, T., Bentley, P.J: Don't push me! Collision-avoiding swarms. In: *Proceedings of the IEEE congress on evolutionary computation*, pp. 1691–1696 (2002)
13. Robinson, J., Sinton, S., Rahmat-Samii, Y.: Particle swarm, genetic algorithm, and their hybrids: optimization of a profiled corrugated horn antenna. In: *IEEE swarm intelligence symposium*, pp. 314–317 (2002)
14. Hendtlass, T.: A combined swarm differential evolution algorithm for optimization problems. In: *Lecture notes in computer science 2070*, pp. 11–18.(2001)
15. Poli, R., Kennedy, J., Blackwell, T.: Particle swarm optimization. *Swarm Intelligence* **1**, 33–57 (2007)
16. Sun, J., Xu, W.B., Feng, B.: Particle swarm optimization with particles having quantum behavior. In: *Proceedings Congress on Evolutionary Computation*, pp. 325–331 (2004)
17. Sun, J., Xu, W.B., Feng, B.: A global search strategy of quantum behaved particle swarm optimization. In: *Proceedings IEEE Conference on Cybernetics and Intelligent Systems*, pp. 111–116 (2004)
18. Sun, J., Xu, W.B., Fang, W.: Quantum-behaved particle swarm optimization with a hybrid probability distribution. In: *proceeding of 9th Pacific Rim International Conference on Artificial Intelligence* (2006)
19. Liu, J., Sun, J., Xu, W.B.: Improving quantum-behaved particle swarm optimization by simulated annealing, *LNAI 4203*, pp. 77–83. Springer, Italy (2006)
20. Coelho, L.S.: Novel Gaussian quantum-behaved particle swarm optimizer applied to electromagnetic design. *Sci, Meas Technol.* **1**, 290–294 (2007)
21. Liu, J., Sun, J., Xu, W.B.: Quantum-behaved particle swarm optimization with immune memory and vaccination. In: *Proceedings IEEE International conference on Granular Computing, USA*, pp. 453–456 (2006)
22. Angeline, P.J.: Using selection to improve particle swarm optimization. In: *Proceedings 1998 IEEE International Conference on Evolutionary Computation. Piscataway*, pp. 84–89 (1998)
23. Shi, Y., Eberhart, R.C.: A modified particle swarm. In: *Proceedings 1998 IEEE International Conference on Evolutionary Computation, Piscataway*, pp. 69–73 (1998)
24. Clerc, M., Kennedy, J.: The particle swarm: explosion, stability, and convergence in a multi-dimensional complex space. *IEEE Trans. Evol. Comput., Piscataway* **6**, 58–73 (2002)
25. Shi, Y., Eberhart, R.: Empirical study of particle swarm optimization. In: *Proceedings 1999 Congress on Evolutionary Computation, Piscataway*, pp. 1945–1950 (1999)