

Quantum-Behaved Particle Swarm Optimization Based on Diversity-Controlled

Haixia Long, Haiyan Fu, Chun Shi

► **To cite this version:**

Haixia Long, Haiyan Fu, Chun Shi. Quantum-Behaved Particle Swarm Optimization Based on Diversity-Controlled. 13th Conference on e-Business, e-Services and e-Society (I3E), Nov 2014, Sanya, China. pp.132-143, 10.1007/978-3-662-45526-5_13 . hal-01342138

HAL Id: hal-01342138

<https://hal.inria.fr/hal-01342138>

Submitted on 5 Jul 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Quantum-behaved particle swarm optimization based on Diversity-controlled

HaiXia Long¹, HaiYan Fu², Chun Shi³

^{1,2,3}School of Information Science Technology, Hainan Normal University, Haikou 571158, Hainan, China

¹haixia_long@163.com

²48318159@qq.com

³605515770@qq.com

Abstract. Quantum-behaved particle swarm optimization (QPSO) algorithm is a global convergence guaranteed algorithms, 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-controlled into QPSO (QPSO-DC) to enhance the diversity of particle swarm, and then improve the search ability of QPSO. The experiment results on benchmark functions show that QPSO-DC has stronger global search ability than QPSO and standard PSO.

Keywords: global convergence; quantum-behaved particle swarm optimization; diversity-controlled; 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 F. 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.

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]; dis-

persion between particles that are too close to one another [11]; reduction of the attraction of the swarm center to prevent the particles clustering too tightly in one region of the search space [12]; hybrids with other meta-heuristic such as genetic algorithms [13]; or ant colony optimization [14]; an up-to-date overview of the PSO [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 decrease 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 by using the characteristic of the problem to guide the search process [21].

In this paper, QPSO with diversity-controlled is introduced. This strategy is to prevent the diversity of particle swarm declining in the search of later stage. The rest of the paper is organized as follows. In Section 2, the principle of the PSO is introduced. The concept of QPSO is presented in Section 3 and the QPSO with diversity-controlled is proposed in Section 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 num-

bers 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

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

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

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

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 move in the search space with a δ 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 δ 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. (2004b) 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

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

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 are 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 α 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 equation (9) Quantum-behaved Particle Swarm Optimization (QPSO), where parameter α must be set as $\alpha < 1.782$ to guarantee convergence of the particle [16]. In most cases, α decrease linearly from α_0 to α_f ($\alpha_0 < \alpha_f$).

We outline the procedure of the QPSO algorithm as follows:

Procedure of the QPSO algorithm:

Step1: Initialize the population;

Step2: Computer the personal position and global best position;

Step3: Computer the mean best position C;

Step4: Properly select the value of α ;

Step 5: Update the particle position according to Eq. (9);

Step6: While the termination condition is not met, return to step (2);

Step7: Output the results.

4 QPSO with diversity-controlled

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. 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, to late stage of search, because the particles are gathered in a relatively small range, particles swarm diversity is very low, the global search ability becomes very weak, the ability for a large range of search has been very small, this algorithm will occur the phenomenon of premature.

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

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

$$\text{diversity(pbest)} = \frac{1}{M \cdot |A|} \cdot \sum_{i=1}^M \sqrt{\sum_{j=1}^D (\text{pbest}_{i,j} - \overline{\text{pbest}})^2} \quad (10)$$

where M is the population of the particle, $|A|$ is the length of longest the diagonal in the search pace, 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 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) Coeffi-

cient. On the course of evolution, if the diversity measure diversity (pbest) of the swarm drops to below the low bound d_{low} , the mean best position is reinitialized.

5 Experiment Results and Discussion

To test the performance of the QPSO with diversity-controlled, seven 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 zeros. The initial range of the 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 maximum number of generations is set as 1000, 1500, and 2000 corresponding to the dimensions 10, 20, and 30 for first six functions, respectively. The maximum generation for the last function is 2000. In order to investigate whether the QPSO-RS algorithm well or not, different population sizes are used for each function with different dimension. They are population sizes of 20, 40, and 80. For SPSO, the acceleration coefficients are set to be $c1=c2=2$ 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-DC, the value of CE Coefficient is listed in Table 3 and Table 4. From the Table 3 and Table 4, we also obtain the CE COEFFICIENT of QPSO-DC decreases from 0.8 to 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 Table 5 to Table 11.

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	$f_5 = 20 + e - 20e^{-\frac{1}{5}\sqrt{\frac{1}{n}\sum_{i=1}^n x_i^2}} - e^{-\frac{1}{n}\sum_{i=1}^n \cos(2\pi x_i)}$	$-30 \leq x_i \leq 30$
Schwefel	$f_6 = 418.9829 \cdot n - \sum_{i=1}^n (x_i \sin \sqrt{ x_i })$	$-500 \leq x_i \leq 500$

Shaffer's	$f_7(X) = 0.5 + \frac{(\sin(\sqrt{x^2 + y^2}))^2}{(1.0 + 0.001(x^2 + y^2))^2}$	$-100 \leq x_i \leq 100$
-----------	--	--------------------------

Table 2. The initial range of population for all the tested algorithms and Vmax for SPSO

	Initial Range	Vmax
f_1	(50, 100)	100
f_2	(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
f_7	(30, 100)	100

Table 3. Parameter value of QPSO-dc

CE Coefficient	Sphere function		Rosenbrock function		Rastrigrin function	
	fitness	St. Dev.	fitness	St. Dev.	fitness	St. Dev.
(1.0,0.5)	1.0050e-013	2.8029e-013	63.6787	112.2467	96.4876	11.2102
(0.9,0.5)	2.1644e-017	3.3232e-017	53.2366	75.9345	94.8874	11.2350
(0.8,0.5)	4.9846e-026	1.7413e-025	39.2303	37.6453	89.7231	9.9548
(0.7,0.5)	9.7934e-022	3.6896e-021	73.4979	100.0602	83.4991	11.6756
(1.0,0.4)	4.6673e-012	2.4765e-011	48.4907	46.6895	81.5502	13.5262
(0.9,0.4)	1.9889e-014	7.9829e-014	78.9340	124.4234	80.8902	12.1062
(0.8,0.4)	6.0362e-015	3.4761e-014	72.1817	102.6235	74.1639	10.4042
(0.7,0.4)	1.6104e-018	4.5714e-018	102.6623	155.5243	70.7840	15.0399
(1.0,0.3)	1.7248e-010	4.4884e-010	59.0231	74.3503	64.9413	19.3783
(0.9,0.3)	1.0957e-011	6.1253e-011	76.6616	103.9478	59.4859	17.9470
(0.8,0.3)	6.2854e-014	1.2280e-013	80.0582	91.4265	56.3751	17.4575
(0.7,0.3)	2.4054e-013	9.3876e-013	106.6513	133.7599	48.8128	20.5664

Table 4. Table 3 (continue)

CE Coefficient	Griewank function		Ackley function		Schwefel function		Shaffer's f6 function	
	fitness	St. Dev.	fitness	St. Dev.	fitness	St. Dev.	fitness	St. Dev.
(1.0,0.5)	0.0219	0.0254	3.0209e-013	4.3354e-013	5.0840e+003	238.6654	7.7802e-004	0.0027
(0.9,0.5)	0.0165	0.0171	3.2649e-014	8.3201e-014	4.9791e+003	218.1045	5.8366e-004	0.0023
(0.8,0.5)	0.0078	0.0096	1.0978e-014	9.2924e-015	3.9491e+003	682.3425	7.8734e-004	0.0027
(0.7,0.5)	0.0141	0.0183	3.9826e-014	5.6935e-014	4.4273e+003	347.6864	0.0019	0.0039
(1.0,0.4)	0.0131	0.0170	3.7380e-011	1.3192e-010	5.0548e+003	251.4751	5.8369e-004	0.0023
(0.9,0.4)	0.0182	0.0191	1.5916e-	3.8720e-	5.0319e+003	238.8422	7.9738e-	0.0027

			012	012			004	
(0.8,0.4)	0.0096	0.0106	8.0437e-013	1.2495e-012	4.8485e+003	267.2251	0.0016	0.0036
(0.7,0.4)	0.0130	0.0175	0.0231	0.1634	4.3537e+003	513.4628	0.0027	0.0044
(1.0,0.3)	0.0139	0.0143	3.9977e-010	7.1079e-010	4.9716e+003	262.9720	1.0843e-007	2.0595e-007
(0.9,0.3)	0.0129	0.0207	8.3162e-011	1.1640e-010	4.9157e+003	242.6659	7.9009e-004	0.0027
(0.8,0.3)	0.0148	0.0141	1.4957e-010	3.6527e-010	4.7433e+003	334.3465	0.0017	0.0038
(0.7,0.3)	0.0142	0.0123	0.0231	0.1634	4.8584e+003	273.0154	0.0027	0.0044

Table 5. Numerical results on Sphere function

M	Dim.	Gmax	SPSO		QPSO		QPSO-DC	
			fitness	St. Dev.	fitness	St. Dev.	fitness	St. Dev.
10	1000	4.6119e-021	1.0352e-020	3.1979e-043	2.2057e-042	5.2719e-045	1.5684e-045	
20	20	1500	9.0266e-012	3.5473e-011	1.9197e-024	7.1551e-024	2.5427e-026	4.2876e-026
	30	2000	3.9672e-008	7.0434e-008	7.3736e-015	2.1796e-014	3.2486e-016	5.3971e-016
40	10	1000	1.3178e-024	3.7737e-024	2.7600e-076	1.8954e-075	8.4527e-074	2.3460e-074
	20	1500	2.3057e-015	6.6821e-015	3.9152e-044	1.8438e-043	5.4381e-045	8.5426e-045
	30	2000	1.1286e-010	2.2994e-010	2.6424e-031	6.3855e-031	2.5419e-032	1.3645e-032
80	10	1000	1.6097e-028	7.1089e-028	2.5607e-103	6.4847e-103	1.9824e-104	2.5681e-104
	20	1500	6.3876e-018	1.4821e-017	1.4113e-068	7.4451e-068	1.5873e-068	4.3876e-068
	30	2000	3.2771e-013	7.5971e-013	6.5764e-050	3.6048e-049	2.0254e-050	1.2574e-050

Table 6. Numerical results on Rosenbrock function

M	Dim.	Gmax	SPSO		QPSO		QPSO-DC	
			fitness	St. Dev.	fitness	St. Dev.	fitness	St. Dev.
20	10	1000	51.0633	153.7913	9.5657	16.6365	7.5341	0.8634
	20	1500	100.2386	140.9822	82.4294	138.2429	32.8419	28.3476
	30	2000	160.4400	214.0316	98.7948	122.5744	69.3102	54.3791
40	10	1000	24.9641	49.5707	8.9983	17.8202	4.9810	3.2094
	20	1500	59.8256	95.9586	40.7449	41.1751	15.3429	12.3076
	30	2000	124.1786	269.7275	43.5582	38.0533	35.2487	24.8619
80	10	1000	19.0259	41.6069	6.8312	0.3355	4.1086	3.5476
	20	1500	40.2289	46.8491	33.5287	31.6415	15.6271	0.0367
	30	2000	56.8773	57.8794	44.5946	31.6739	22.3706	0.1648

Table 7. Numerical results on Rastrigrin function

M	Dim.	Gmax	SPSO		QPSO		QPSO-DC	
			fitness	St. Dev.	fitness	St. Dev.	fitness	St. Dev.
20	10	1000	5.8310	2.5023	4.0032	2.1409	2.6871	1.9684
	20	1500	23.3922	6.9939	15.0648	6.0725	13.5970	12.3458
	30	2000	51.1831	12.5231	28.3027	12.5612	22.3694	4.2061
40	10	1000	3.7812	1.4767	2.6452	1.5397	1.8541	2.1380
	20	1500	18.5002	5.5980	11.3109	3.5995	12.3471	7.2684
	30	2000	39.5259	10.3430	18.9279	4.8342	14.3796	6.9173
	10	1000	2.3890	1.1020	2.2617	1.4811	1.0367	1.3574

80	20	1500	12.8594	3.6767	8.4121	2.5798	6.3247	4.6873
	30	2000	30.2140	7.0279	14.8574	5.0408	13.6975	5.3671

Table 8. Numerical results on Griewank function

M	Dim.	Gmax	SPSO		QPSO		QPSO-DC	
			fitness	St. Dev.	fitness	St. Dev.	fitness	St. Dev.
20	10	1000	0.0920	0.0469	0.0739	0.0559	0.0657	0.0502
	20	1500	0.0288	0.0285	0.0190	0.0208	0.0262	0.0227
	30	2000	0.0150	0.0145	0.0075	0.0114	0.0082	0.0235
40	10	1000	0.0873	0.0430	0.0487	0.0241	0.0587	0.0822
	20	1500	0.0353	0.0300	0.0206	0.0197	0.0135	0.0431
	30	2000	0.0116	0.0186	0.0079	0.0092	0.0020	0.0123
80	10	1000	0.0658	0.0266	0.0416	0.0323	0.0424	0.0682
	20	1500	0.0304	0.0248	0.0137	0.0135	0.0050	0.0103
	30	2000	0.0161	0.0174	0.0071	0.0109	1.0570e-006	5.3089e-006

Table 9. Numerical results on Ackley function

M	Dim.	Gmax	SPSO		QPSO		QPSO-DC	
			fitness	St. Dev.	fitness	St. Dev.	fitness	St. Dev.
20	10	1000	2.0489e-011	3.0775e-011	6.8985e-012	1.2600e-011	2.7356e-015	5.0243e-016
	20	1500	0.0285	0.2013	1.5270e-008	2.1060e-008	9.7700e-015	5.8751e-015
	30	2000	0.2044	0.4899	4.3113e-007	4.4188e-007	7.1797e-013	1.5299e-012
40	10	1000	2.4460e-013	5.2901e-013	2.5935e-015	5.0243e-016	2.6645e-015	0
	20	1500	2.6078e-008	4.0653e-008	6.3491e-013	1.3305e-012	3.5882e-015	1.5742e-015
	30	2000	3.7506e-006	6.4828e-006	5.5577e-011	8.6059e-011	6.5015e-015	1.8772e-015
80	10	1000	5.5778e-015	7.4138e-015	2.4514e-015	8.5229e-016	2.2382e-015	1.1662e-015
	20	1500	5.2979e-010	7.2683e-010	5.8620e-015	1.6446e-015	2.7356e-015	5.0243e-016
	30	2000	1.6353e-007	2.1300e-007	6.7253e-014	4.4759e-014	5.4357e-015	1.4866e-015

Table 10. Numerical results on Schwefel function

M	Dim.	Gmax	SPSO		QPSO		QPSO-DC	
			fitness	St. Dev.	fitness	St. Dev.	fitness	St. Dev.
20	10	1000	630.6798	169.9164	1.0784e+003	342.1967	740.4763	293.0045
	20	1500	2.0827e+003	327.0313	4.0026e+003	679.1461	3.4182e+003	1.0382e+003
	30	2000	4.1097e+003	481.8801	6.5075e+003	589.9931	6.8196e+003	1.1614e+003
40	10	1000	494.0236	169.5594	1.4359e+003	246.9290	876.2897	382.1407
	20	1500	1.8516e+003	273.8690	4.5580e+003	242.8220	3.9461e+003	816.6767
	30	2000	3.4042e+003	384.3150	7.3380e+003	288.9463	7.3466e+003	879.8090
	10	1000	426.0796	130.5357	1.6060e+003	132.0009	979.6775	363.0876

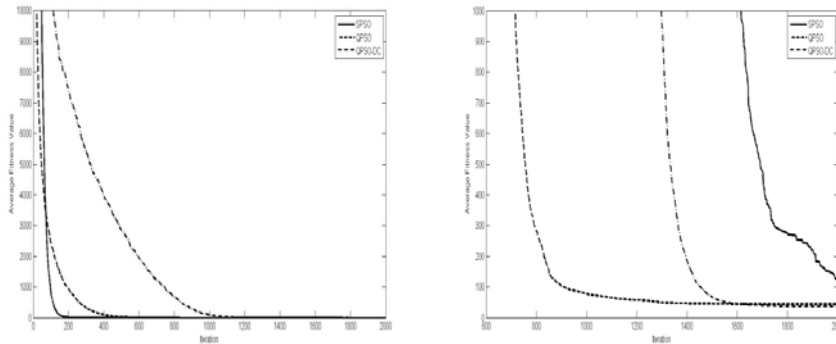
80	20	1500	1.5466e+003	270.5587	4.6407e+003	225.9431	4.1601e+003	518.4552
	30	2000	3.0325e+003	419.8580	8.0583e+003	226.3796	7.3655e+003	852.6205

Table 11. Numerical results on Shaffer's f6 function

M	Dim.	Gmax	SPSO		QPSO		QPSO-DC	
			fitness	St. Dev.	fitness	St. Dev.	fitness	St. Dev.
20	2	2000	3.8965e-004	0.0019	0.0016	0.0036	1.3764e-007	7.3542e-007
40	2	2000	1.9432e-004	0.0014	5.8308e-004	0.0023	1.2681e-008	3.6984e-008
80	2	2000	0.0000	0.0000	7.5695e-009	2.5080e-008	5.9540e-009	8.0367e-009

The results show that both QPSO, QPSO-DC are superior to SPSO except on Schwefel and Shaffer's f6 function. On Shpere Function the QPSO works better than QPSO-DC 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 instance, the best result is QPSO-DC. The Rosenbrock function is a mono-modal function, but 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-DC outperforms the QPSO. Rastrigrin function and Griewank function are both multi-modal and usually tested for comparing the global search ability of the algorithm. On Rastrigrin function, it is also shown that the QPSO-DC generated best results than QPSO. On Griewank function, QPSO-DC has better performance than QPSO algorithm. On Ackley function, QPSO-DC has best performance when the dimension is 10, except for above functions, the QPSO-DC has minimal value. On the Schwefel function, SPSO is best performance in any situation. On Shaffer's function, it has sub-optima 0.0097. The QPSO-DC shows its stronger ability to escape the local minima 0.0097 than the QPSO, and even than SPSO. Generally speaking, the QPSO-DC 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-DC converge more slowly than the QPSO 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.



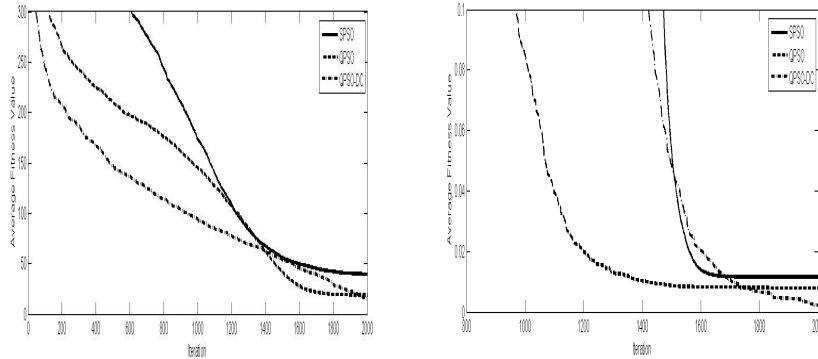


Figure1. Convergence process of the three algorithm on the first four benchmark functions with dimension 30 and swarm size 40 averaged on 50 trail runs

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

Aknowledgements.

This work was financially supported by the National Natural Science Fund (No. 71461008, No. 61362016) and the Hainan Province Natural Science Fund (No. 614235), and the Higher School Scientific Research Project of Hainan Province (Hjkj2013-22).

References

1. Kennedy, J, Eberhart, R, Particle swarm optimization, In: Proc.IEEE Int. Conf. Neural Networks, 1995, pp.1942–1948. F. Van den Bergh, An Analysis of Particle Swarm Optimizers. University of Pretoria, South Africa, 2001.
2. Clerc, M, Discrete particle swarm optimization illustrated by the traveling salesman problem. New optimization techniques in engineering, Berlin: Springer; 2004, pp. 219–239.
3. Clerc, M, Particle swarm optimization, ISTE, 2006.
4. Zhang, W., Liu, Y., Clerc, M. An adaptive PSO algorithm for reactive power optimization, In: Sixth international conference on advances in power control, operation and management, Hong Kong, 2003.
5. Eberhart, R. C., Shi, Y. Comparing inertia weights and constriction factors in particle swarm optimization, In: IEEE international conference on evolutionary computation, 2001, pp. 81–86.
6. He, S., Wu, Q. H., Wen, J. Y. Saunders J R, Paton R C, A particle swarm optimizer with passive congregation. Biosystems 78, 2004, 135–47.
7. Hu ,X., Eberhart, R. C. Tracking dynamic systems with PSO: where’s the cheese?, In: Proceedings of the workshop on particle swarm optimization, Indianapolis, USA, 2001.
8. Parsopoulos, K. E., Vrahatis, M. N. Particle swarm optimizer in noisy and continuously changing environments, Artificial intelligence and soft computing , 2001, 289–294.

9. Parsopoulos, K. E., Vrahatis, M. N. On the computation of all global minimizers through particle swarm optimization, *IEEE Transactions on Evolutionary Computation* 8 , 2004, 211–24.
10. LoZvbjerg, M., Krink, T. Extending particle swarms with self-organized criticality, In: *Proceedings of the IEEE congress on evolutionary computation*, 2002, pp. 1588–1593.
11. Blackwell, T., Bentley, P. J. Don't push me! Collision-avoiding swarms, In: *Proceedings of the IEEE congress on evolutionary computation*, 2002, pp. 1691–1696.
12. 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*, 2002, pp. 314–317.
13. Hendtlass, T. A combined swarm differential evolution algorithm for optimization problems, In: *Lecture notes in computer science 2070 (2001)* 11–18.
14. Poli, R., Kennedy, J. Blackwell T, Particle swarm optimization, *Swarm Intelligence* 1 , 2007, 33-57.
15. Sun, J., Xu, W. B. Feng B, Particle swarm optimization with particles having quantum behavior, *Proc. Congress on Evolutionary Computation*, 2004, pp. 325-331.
16. Sun, J., Xu, W. B. Feng B, A global search strategy of quantum behaved particle swarm optimization, *Proc. IEEE Conference on Cybernetics and Intelligent Systems*, 2004, pp. 111-116.
17. Sun, J., Xu, W. B. Fang W, Quantum-behaved particle swarm optimization with a hybrid probability distribution, In the proceeding of 9th Pacific Rim International Conference on Artificial Intelligence, 2006.
18. Liu, J., Sun, J., Xu, W. B. Improving quantum-behaved particle swarm optimization by simulated annealing, *LNAI 4203*, Springer, Italy, 2006, pp. 77-83.
19. Coelho, L. S. Novel Gaussian quantum-behaved particle swarm optimizer applied to electromagnetic design, *Science, Measurement & Technology* 1 , 2007, 290-294.
20. Liu, J., Sun, J., Xu, W. B. Quantum-behaved particle swarm optimization with immune memory and vaccination, In: *Proc. IEEE International conference on Granular Computing, USA*, 2006, pp. 453-456.
21. Angeline, P. J. Using Selection to Improve Particle Swarm Optimization, In: *Proc. 1998 IEEE International Conference on Evolutionary Computation*. Piscataway, NJ, 1998, pp. 84-89.
22. Shi, Y., Eberhart, R.C. A Modified Particle Swarm, In: *Proc. 1998 IEEE International Conference on Evolutionary Computation*, Piscataway, NJ, 1998, pp. 69-73.
23. Clerc, M., Kennedy, J. The Particle Swarm: Explosion, Stability, and Convergence in a Multi-dimensional Complex Space, *IEEE Transactions on Evolutionary Computation*, Piscataway, NJ, 6 , 2002, 58-73.
24. Shi, Y. Eberhart R, Empirical Study of Particle Swarm Optimization, in: *Proc. 1999 Congress on Evolutionary Computation*, Piscataway, NJ, 1999, pp. 1945-1950.