Quantum-behaved Particle Swarm Optimization Algorithm Based on Dynamic Dual-population Joint-search Mechanism

Jinling Niu, Yong Zhang, Zhongfeng Li

***Abstract*—Quantum particle swarm optimization (QPSO) has disadvantages such as rapid loss of species diversity and inability to jump out of local optimum value in the later stage. In this paper, a QPSO algorithm based on dynamic dual-population joint-search mechanism (DJ-QPSO) is proposed. This algorithm establishes two local attraction points in the search area to guide the particle search in the population, and adjusts the global exploration and local exploitation ability by changing the population diversity. Then, the algorithm uses a periodic dynamic-sharing strategy to enable information exchange between the two subgroups. Finally, a global convergence formula is introduced to the search in the later stage to improve algorithm precision. The simulation results of 15 benchmark functions demonstrate that the improved algorithm performs better than comparable algorithms and can effectively deal with complex optimization problems.**

***Keywords*—Dynamic dual-population joint, DJ-QPSO, Diversity analysis, Periodic sharing strategy**

I. Introduction

Quantum particle swarm optimization (QPSO) algorithm is a PSO variant algorithm proposed from the perspective of quantum mechanics. Sun [1] proposed it after studying the results of particle convergence behavior reported by Clerc et al [1]. The QPSO algorithm considers that the particle has quantum behavior. Particle trajectory analysis indicates that the algorithm takes the local attraction point as attractor and causes the population to remain aggregated by the attraction point. However, the single local attraction causes the particles to fall easily into local optimum and restricts the global search ability.

To address the defects of QPSO, many domestic and foreign scholars have proposed various improved methods. The main improvement directions are control parameter design, improved algorithm strategy, and multiple algorithm fusion. Li [2] proposed an improved QPSO algorithm, which uses Monte Carlo method to first obtain the particles of multiple individuals, which then cooperate with one another. Experimental results show that CQPSO is superior to other improved QPSO in terms of solving quality and computation cost. Inspired by the classical particle swarm method and quantum mechanics theories, Coelho [3] presented a new quantum-behaved approach using a mutation operator with exponential probability distribution; the simulation results demonstrate good performance of the proposed algorithm to solve a significant benchmark problem in electromagnetic devices. Liu [4] proposed a new cultural algorithm and introduced cultural evolution mechanism to study QPSO for multi-objective problems. In cultural MOQPSO, the exemplar positions of each particle are obtained according to "belief space," which contains various types of knowledge. Moreover, to increase population diversity and obtain continuous and evenly distributed Pareto fronts, a combination-based update operator is proposed to update the external population in the study. The effectiveness of the algorithm is verified by an example. Turgut [5] proposed a novel chaotic quantum-behaved PSO algorithm to solve nonlinear system of equations. Different chaotic maps are introduced to enhance the effectiveness and robustness of the algorithm. Comparison results reveal that the proposed algorithm can cope with the highly nonlinear problems and outperforms many algorithms presented in the literature. Huang [6] proposed a QPSO algorithm with adaptive inertia weight adjustment, introduces the evolution velocity factor s_d and aggregation degree factor j_d of the QPSO, adjusts the inertial weight through these two dynamic parameters, and significantly improves the convergence speed of the algorithm. Wang [7] proposed a QPSO algorithm with Gauss perturbation in the average or global optimal position of particles, which can effectively prevent the stagnation of particles and prevent particles from falling into local optimum; the results show that the algorithm has strong global search ability and fast convergence speed. Wu [8] proposed an improved QPSO algorithm based on random evaluation strategy. The algorithm uses stochastic factors to evaluate the innovation of particles and improves the ability to eliminate local optimum. Fixed value strategy and linear decreasing strategy are proposed to control the unique parameters of the QPSO algorithm. Chen [9] proposed a QPSO algorithm with crossover operator. A new method of calculating the particle attraction point and characteristic length of potential well is used. The crossover operator in genetic algorithm is introduced and the crossover probability adaptive parameter control technology is incorporated to ensure the diversity of particle swarm and maintain the vitality of the whole particles. Experiments show the effectiveness and robustness of the improved algorithm. Zhang [10] proposed a QPSO algorithm based on two elite

This work was supported in part by the National Nature Science Foundation of China (61473054).

Yong Zhang (corresponding author) is with the School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, CO 114051, China. (e-mail: zy9091@163.com).

Jinling Niu (author) is with the School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, CO 114051, China. (e-mail:1443868965@qq.com).

Zhongfeng Li (author) is with the Yingkou Institute of Technology. (e-mail: afeng0601@163.com).

learning strategies. The elite particles are searched by dynamic approximation search strategy to avoid falling into local optimum and provide good guidance for the population. The disturbance is used for large-scale exploration to ensure that the algorithm has higher global search performance. The experimental results of benchmark functions show that the method has good global convergence and high search accuracy. Wei [11] proposed a quantum behavioral particle swarm optimization (CLQPSO) algorithm based on comprehensive learning strategy, which changes the updating method of local attractors in QPSO and fully utilizes social information in the group. The performance of CLQPSO is tested by several benchmark functions. The experimental results show that the algorithm can find an improved solution. Jia [12] proposed a niche particle swarm optimization (NCPSO) algorithm based on chaotic mutation. The algorithm combines niche technology with elimination mechanism, which causes the algorithm to have good global optimization ability, and the variable scale chaotic mutation has fine local traversal search performance, which ensures that the algorithm has high search accuracy. The experimental results show that the algorithm has the advantages of strong optimization ability, high search accuracy, and good stability. Wang [13] proposed a two-population QPSO algorithm based on chaos optimization. The two populations generated by chaotic sequences have their own updating strategies, and they interact with each other through the fusion of population changes. The convergence speed and traversal range of the algorithm are improved. Furthermore, binary coded QPSO algorithms [14], diversity control QPSO algorithms [15], improved artificial bee colony search operator quantum PSO algorithm, and others are also available [16].

These proposed algorithms have improved the optimization algorithm performance to a certain extent, but the fundamental limitation of the QPSO algorithm is that the global and local searches restrict each other. Thus, in this paper, we propose a QPSO algorithm based on dynamic dual-population search to balance global exploration ability and local exploitation capability.

The rest of the paper is organized as follows. Section 2 introduces the basic principle of standard QPSO, Section 3 describes the basic principle of DJ-QPSO algorithm, including the setting of double attraction points, sharing strategy, Gaussian chaotic mutation and global convergence formula. Section 4 tests the benchmark functions and result analysis. Section 5 provides the conclusion and future research directions.

II. QPSO ALGORITHM

The QPSO algorithm combines standard PSO and quantum mechanics. In quantum mechanics, the movement of quantum particles to the lowest point of potential energy in the potential field is equivalent to the optimization of particles. The potential well in quantum mechanics is equivalent to the optimum range of particles, and the lowest potential energy in the potential field is the global optimal solution [1].

The QPSO algorithm attracts particles in the population by establishing a potential well at the local attraction point $pi = (p_{i,1}, p_{i,2}, \ldots, p_{i,N})$, where the pi point is

$$p_{i,j}(t) = \frac{c_1 r_{1_{i,j}} P_{i,j}(t) + c_2 r_{2_{i,j}} G_{i,j}(t)}{c_1 r_{1_{i,j}} + c_2 r_{2_{i,j}}}, 1 < j < N \quad (1)$$

where, $P_{i,j}$ represents the j-dimensional component of the particle i's individual optimal value of the D-dimensional search space, and $G_{i,j}$ represents the j-dimensional component of the particle i's global optimal value of the D-dimensional search space. r_1 and r_2 are independent distributed random numbers between [0, 1], which are called random factors, and C_1 and C_2 are acceleration coefficients.

The evolution equation of particles in QPSO is:

$$X_{i,j}(t+1) = p_{i,j}(t) \pm \alpha \left| C_j(t) - X_{i,j}(t) \right| \ln\left[1/u_{i,j}(t) \right] \quad (2)$$

where $u_{i,j}(t)$ is uniformly distributed between [0,1], and α is called contraction–expansion coefficient, which has two control modes: fixed value and linear reduction. $L_{i,j}$ is the potential well length, and the method of evaluation is

$$L_{i,j}(t) = 2\alpha \left| C_j(t) - X_{i,j}(t) \right| \quad (3)$$

We introduce an average value $C(t)$ representing the best position of all particle individuals, as defined in the following:

$$C(t) = \frac{1}{N} \sum_{i=1}^{M} p_i(t) \quad (4)$$

The updating methods of individual optimal value P_i and global extreme G_i of particles are exactly the same as those of the standard PSO algorithm.The updated formula is:

$$P_i(t+1) = \begin{cases} P_i(t), & f(X_i(t+1)) \geq f(P_i(t)) \\ X_i(t+1), & f(X_i(t+1)) \geq f(P_i(t)) \end{cases} \quad (5)$$

$$P_g(t+1) = \arg\min f(P_i(t+1)), 1 \leq i \leq N \quad (6)$$

The preceding equation expresses the basic principle of QPSO, using which an improved method is proposed in this paper.

III. QUANTUM-BEHAVED PSO ALGORITHM BASED ON DYNAMIC DUAL-POPULATION JOINT-SEARCH MECHANISM

The setting of local attraction points and bound motion of quantum particles in QPSO enables the algorithm to have good convergence, but at the same time causes the loss of population diversity too early and too fast. The algorithm converges quickly and has a large probability of falling into local optimum value. In this paper, a dynamic dual-population joint-search mechanism is introduced to QPSO. Two local attraction points are established to divide the population into two parts and guide the search. Periodic dynamic interaction strategy is used to exchange information between subgroups. Finally, the formula of global convergence is introduced, and the fine search is performed near the global optimal solution. When the global search is guaranteed, the large probability of the algorithm converges to the global optimal value.

A. Exploitation and exploration subgroups

In QPSO, the average best position C and individual learning tendency point P are used to evaluate the particles. DJ-QPSO evenly divides the population particles into two subgroups. The two subgroups are attracted by the attraction points for searching. The particle swarm guided by the attraction point *P* of the average best position *C* is the exploitation subgroup *N1*. The subgroup *N1* is assisted by a small step size near the average best position *C*. The subgroup *N2* guided by the local search point *Q* is the exploration subgroup, and the subgroup *N2* is guided by the average best position *D*. The subgroup *N2* searches the entire solution space in a large scale with a large step size, trying to find the possible region of the optimal solution, where *N1* = *N2*. The population number is *N* = *N1*+*N2*. The dual-population joint search not only improves the search precision but also prevents the search particle from falling into local optimum too quickly.

The evaluation method of local attraction point *P* is:

$$p_{i,j}(t)=\frac{c_1 r_{1_{i,j}} P_{i,j}(t)+c_2 r_{2_{i,j}} G_{i,j}(t)}{c_1 r_{1_{i,j}}+c_2 r_{2_{i,j}}}, 1\le i\le N_1 \tag{7}$$

The evolution equation of local attraction point *P* guidance is as follows:

$$X_{i,j}(t+1)=p_{i,j}(t)\pm w\cdot\left|C_j(t)-X_{i,j}(t)\right|\cdot\ln u_i \tag{8}$$

where in QPSO, the value of α can be set by means of a fixed value and a linear reduction control method, and in DJ-QPSO, the inertia factor *w* is introduced to replace the value of α, where

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

With the increase in the number of iterations, the inertia factor *w* decreases gradually. The value of inertia factor *w* in the early stage of the search is larger, which is conducive to global search, thereby increasing the diversity of the population and decreasing the value in the later stage of the search is conducive to local search.

The evaluation method of local search point *Q* is

$$q_{i,j}(t)=\frac{c_1 r_{1_{i,j}} P_{i,j}(t)+c_2 r_{2_{i,j}} R_{i,j}(t)}{c_1 r_{1_{i,j}}+c_2 r_{2_{i,j}}},\quad N_1\le i\le N \tag{10}$$

The evolutionary equation of *Q* guided by local search points is:

$$\begin{aligned} X'_{i,j}(t+1)&=q_{i,j}(t)\pm w\cdot e^{r_1}\sin(2\pi\cdot r_2)\\ &\cdot\left|D_j(\mathrm{t})-X_{i,j}(t)\right| \end{aligned} \tag{11}$$

where $R_{i,j}(t)$ is the individual optimal value of a randomly selected particle; $D_j(t)$ is the average optimal value for exploration subgroup particles, and the evaluation method is the same as $C(t)$.

In the iterative search process, two subgroups adopt different optimization strategies. The exploitation subgroup *N1* is mainly responsible for fine search. Around the logarithmic curve, the search step continuously decreases as the number of iterations increases, and can be clustered near the optimal value. Exploration subgroup *N2* is responsible for large-scale rough search, and the position update formula is a sinusoidal exponential function. As the search step of the optimization process increases, it can be developed globally in new fields.

B. Periodic information sharing strategy

In the optimization process, the two subgroups are searched independently, and each of them takes the subgroup optimal solution as the goal to make decisions. To improve the global optimization ability, the two subgroups must establish a good information sharing mechanism.

In the optimization process of QPSO, the position information of individual optimum value and global optimum value is updated with each iteration. That is, the best location currently searched will soon be replaced by a new location, which cannot play its role. When the particle is trapped in the local optimal value, the algorithm can hardly continue to search, thereby greatly affecting the global search ability of the algorithm [17].

To solve this problem, the DJ-QPSO algorithm uses periodic update strategy. The periodic update strategy contains a fault-tolerant mechanism, which allows or tolerates a fault occurrence within limits, that is, it allows to miss some of the optimal positions during the current period and continues to be optimal in the previous week. The specific sharing mechanism is such that the two subgroups do not exchange positions after an iteration; however, they share positions after an iteration cycle. To a certain extent, the algorithm not only fully utilizes the best historical position to search and guide but also enables the particles to find the optimal value by sharing the best global position. This periodic updating strategy can improve the global search performance of the algorithm [18][19].

The setting of the sharing period C_0 should also consider its rationality. The sharing period is too small to give full play to the ideological advantages, and the sharing period is too large to miss the optimal location of the sharing, thereby affecting the convergence performance of the algorithm.

C. Gaussian chaotic mutation

The exploitation subgroups *N1* and *N2* share the optimal value information after sharing period C_0. If the exploitation subgroup *N1* falls into the local optimal value in one iteration cycle, it will not be able to provide highly effective optimal value information. Therefore, when the exploitation subgroup *N1* has been iterated for *m* (*m* = 7) [20] consecutive times, the optimal value of the subgroup is not updated, and then the Gaussian chaotic mutation operation is performed, and the Gaussian function is introduced into the position update formula. It can adjust the trajectory of the particles close to the global optimal position *Gbest* in subgroup *N1* by a small margin, and adjust the trajectory of the particles far from *Gbest* by a large margin. Chaotic mapping can re-map the particles to the search space and continue to optimize by calculating logical chaotic mapping, and the omnidirectional ergodicity of the particle motion can be achieved. After the position of the particle is updated, the trajectory is adjusted by using the Gauss function. The formula is as follows:

$$x_i(t+1)=\begin{cases} x_i(k)+\alpha\cdot h_i(k)\cdot \\ \left\|pbest(k)-x_i(k)\right\|_2\cdot(x_{i,\max}-x_i'(t)),\quad \alpha\ge 0 \\ x_i(k)+\alpha\cdot h_i(k)\cdot \\ \left\|pbest(k)-x_i(k)\right\|_2\cdot(x_i'(t)-x_{i,\min}),\quad \alpha\ge 0 \end{cases} \tag{12}$$

where *a* is the trajectory correction coefficient and is a random number between intervals [−1,1], which can ensure that DJ-QPSO searches from positive or negative directions. $h_i(k)$ denotes the introduced Gaussian function. $\|pbest(k)-x_i(k)\|_2$ is the distance between the particle i and the best location.

The Gaussian function $h_i(k)$ is shown as

$$\begin{cases} h_i(t)=1, & pbest_1(t)=x_i(t) \\ h_i(t)=\exp\left(-\dfrac{(pbest_1(t)-x_i(t)^2}{2\sigma^2(t)}\right), & pbest_1(t)\neq x_i(t) \end{cases} \tag{13}$$

$$\sigma'^2(t)=\sigma_0^2 e^{-\frac{t}{\tau_1}} \tag{14}$$

where σ_0 and τ_1 are constants. (13) shows that when this particle locates the best position, the Gaussian function $h_i(k)$ = 1, which means that later modifications do not alter the particle trajectory. Moreover, according to the characteristics of the Gaussian function, the particles close to the global best position adjust further and those far from *Gbest* modify less. Furthermore, according to (13), with the increase in the iterative step t, Gaussian function $h_i(k)$ becomes steeper gradually. Thus, the updating amplitude decreases, guaranteeing the approximation capability of DJ-QPSO in the later period of optimization. In addition, the chaotic map x_i' is introduced into the trajectory modification, as follows:

$$z_j(n+1)=4z_j(n)\cdot(1-z_j(n)) \tag{15}$$

$$x_i'(t)=x_{i,\min}+z_i(t)(x_{i,\max}-x_{i,\min}) \tag{16}$$

where $z_j(n)$ is an iterative mapping parameter of chaos; j and n denote the dimension and index of the particle, respectively; n = 1, 2,…, m and $x_{i,\min}$ and $x_{i,\max}$ are the minimum and maximum positions of the particles. (15) shows a logistic chaotic map, a property of which is 4 for most values of r. A chaotic system exhibits great sensitivity to the initial conditions. Thus, the initial value z is a random number in the interval [0, 1] except for 0.25, 0.5, and 0.75, where the sequence $z_j(n)$ shows the chaotic characteristics, which is better than the uniform distribution in the aspect of travel ergodicity. However, 0.25, 0.5, and 0.75 must be excluded because if these three numbers are chosen as initial values, then $z_j(n)$ will be fixed as 0.75, 0, and 0.75, respectively.

First, the chaotic sequence is generated by (15). Then, the position of particle trajectory is re-mapped to the search space by (16). The ergodicity of the chaotic map is fully utilized so that the precocious particles are re-mapped to the search space after trajectory correction to improve the diversity of the population and ensure the global search ability of the population particles [21][22].

D. Global convergence formula

The setting of double subgroups with double attraction points cannot guarantee the particle convergence to the global optimal value. Thus, we introduced the global convergence formula in the algorithm, and the evaluation method is as follows:

$$P_{i,j}(t+1)=P_{i,j}(t)+\sigma\cdot\varepsilon_t \tag{17}$$

where ε_t obeys the Gaussian normal distribution $N(0,1)$ and σ is the contraction factor, which can control the convergence step and take a fixed value of the control mode. A probability function b is introduced here, which is a typical exponential function

$$\mathrm{b}=e^{-t/T} \tag{18}$$

When Rand >b, the algorithm performs a fine search near the global optimal solution. When Rand <b, the algorithm performs a dual-population search. As the iteration number increases, the value of b decreases gradually, and the probability that the random number is larger than the function value increases. This condition shows that the algorithm is likely to implement the global convergence formula in the later stage of search. In the early stage, the algorithm is likely to implement double population search. By comparing the random number with the probability function, the algorithm chooses between the dual-population search and the global convergence formula.

E. Algorithm steps

The implementation steps of the DJ-QPSO algorithm proposed in this paper are summarized as follows:

Step 1: The parameters are set in the problem solution space, including Dim, swarm population N, exploitation subgroup *N1* and exploration subgroup *N2*, maximum iteration number T, and others.

Step 2: The population is initialized, the initial positions of exploitation subgroup *N1* and exploration subgroup *N2* are set, the fitness of each particle is evaluated, and the position of the least fitness particle is assigned to the optimal position of the subgroup.

Step 3: The average optimal position $C(t)$ and $D(t)$ of two subgroup particles is calculated by (4).

Step 4: The current fitness of the particle is calculated by (8) and (11), and exchange information at the end of sharing period C_0. A comparison of the fitness of the previous iteration cycle, if $F(pbest_{N1})<F(pbest_{N2})$, shows that the global optimal position of the exploitation subgroup N1 is better than that of exploration subgroup *N2* in a sharing period C_0. Then, the particles of exploration subgroup *N2* inherit the global optimal position at this time and continues to search for the next sharing period. If $F(pbest_{N1})>F(pbest_{N2})$, then the global optimal position of the exploration subgroup *N2* is better than that of the exploitation subgroup *N1* in the iteration period, and the exploitation subgroup *N1* adds Gaussian chaotic mutation to research and the exploration subgroup continues to search.

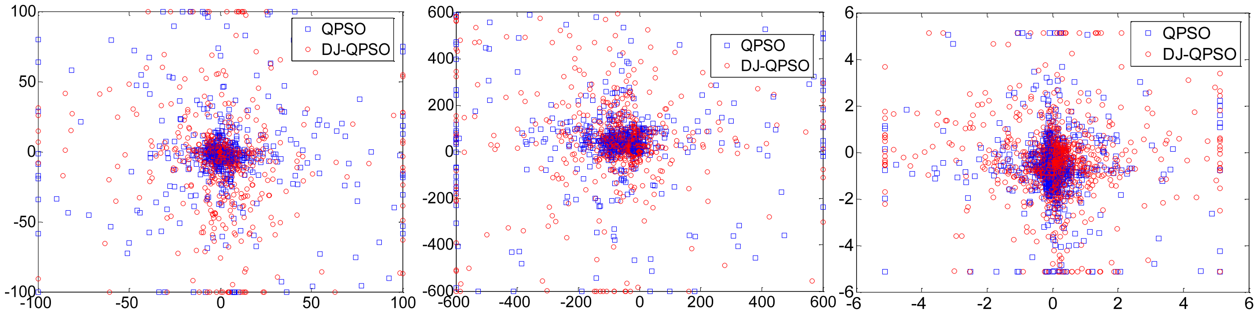
Step 5: The current global optimal location of the group is calculated, that is,

$$G(t)=P_g(t), g=\arg\min_{1\le i\le M}\{F[P_i(t)]\};$$

Step 6: Comparing the current optimal position with the previous iterative global optimal position shows that if the current global optimal position is better, then the global optimal position of the group is updated to its value.

Step 7: The new position of the particle according to (8) and (11).

Step 8: If the maximum number of iterations given is reached, then the search stops. If the termination condition is



(a) Sphere　　(b) Griewank　　(c) Rastrigin

Fig. 1 Diagram of particle distribution state

not met, then the above steps are repeated.

IV. Experimental design and Result analysis

A. Diversity analysis

In the DJ-QPSO algorithm, the diversity of the population can be enhanced by the setting of the double subgroups. To intuitively present in detail the distribution state of the particle diversity in the QPSO and DJ-QPSO, three test functions are selected to analyze the diversity of the algorithm, the specific function expressions are shown in Table I.

The QPSO and DJ-QPSO can be used to solve the multidimensional problem. The population size is 20 and the number of iterations is 100. The particle distribution of the search results is shown in Fig. 1. The red dots in the figure represent the improved DJ-QPSO algorithm, and the blue square points represent QPSO, which depicts the distribution of all the particles in the entire process from the beginning to the end of the iteration [22].

According to the analysis of the particle distribution state diagram, Fig. 1(a) shows a unimodal function *sphere*, in which the two algorithms have the same optimization ability in the solution space. From the range of particle distribution, the two algorithms are roughly the same. From the perspective of search precision, the two algorithms can gather a large number of particles near the optimal value. The precision of the DJ-QPSO algorithm is slightly more accurate, but in general, the performance of the two algorithms is unequal in the unimodal function. In Fig. 1(b), the multimodal function *Griewank* shows that the search precision of the two algorithms is lower than that of the unimodal function, but the search range is wider. The DJ-QPSO algorithm has better search ability than QPSO in terms of search precision and range. In Fig. 1(c), *Rastrigin* is also a multimodal function. The particles of DJ-QPSO have a wider search range in the search area and can traverse the entire solution area. This condition shows that the DJ-QPSO algorithm can help the search particles to jump out of the local optimal value. At the same time, a larger number of particles can be gathered near the global optimal solution. DJ-QPSO has strong convergence ability. We can prove that the DJ-QPSO algorithm can effectively disperse the population and eventually cause a large number of particles to be clustered near the optimal solution, with the ability to balance local exploitation and global exploration.

B. Benchmark function test and analysis

To confirm the validity of the DJ-QPSO algorithm, we select 15 benchmark functions to test its effectiveness. Among these functions, F1–F4 are unimodal and F5–F9 are multimodal. F10–F12 and F13–F15 are rotation and offset functions, respectively. Specific formula expressions and ranges are shown in Table I. The experimental arrangements are as follows: First, the appropriate sharing period C_0 is selected through experiments, and then the appropriate contraction factor σ is determined by experiments. Finally, the DJ-QPSO algorithm is compared with other algorithms when the sharing period C_0 and contraction factor σ are optimal.

TABLE I
BENCHMARK TEST FUNCTIONS

Function name	Function formula	Search space	Optimal value
Sphere	$F_1(X)=\sum_{i=1}^{D}x_i^{\,2}$	[−100,100]	0
Schwefel 2.22	$F_2(X)=\sum_{i=1}^{D}\lvert x_i\rvert+\prod_{i=1}^{D}\lvert x_i\rvert$	[−10,10]	0
Schwefel 1.2	$F_3(X)=\sum_{i=1}^{D}\left(\sum_{j=1}^{i}x_j\right)^2$	[−100,100]	0
Rosenbrock	$F_4(X)=\sum_{i=1}^{D-1}(100(x_{i+1}-x_i^{\,2})^2+(x_i-1)^2)$	[−100,100]	0

Rastrigin	$F_5(X)=\sum_{i=1}^{D}[x_i^{\ 2}-10\cos(2\pi x_i)+10]$	$[-5.12,5.12]$	0
Griewank	$F_6(X)=\sum_{i=1}^{D}\frac{1}{4000}x_i^{\ 2}-\prod_{i=1}^{D}\cos\frac{x_i}{\sqrt{i}}+1$	$[-600,600]$	0
Ackley	$F_7(X)=20+e-20\exp(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D}x_i^{\ 2}})$ $-\exp(\frac{1}{D}\sum_{i=1}^{D}\cos(2\pi x_i))$	$[-32,32]$	0
Schwefel	$F_8(X)=418.9829D+\sum_{i=1}^{D}-x_i\sin(\sqrt{\lvert x_i\rvert})$	$[-500,500]$	0
Generalized Penalized	$F_9(X)=0.1\{\sin^2 3\pi x_1+\sum_{i=1}^{D}(x_i-1)^2[1+\sin^2(3\pi x_{i+1})]\}+$ $(x_D-1)^2[1+\sin^2(2\pi x_D)]+\sum_{i=1}^{D}u(x_i,5,100,4)$ $u(x_i,a,k,m)=\begin{cases}k(x_i-a)^m\ , & x_i>a\\ 0, & -a\le x_i\le a\\ k(-x_i-a)^m\ , & x_i<-a\end{cases}$	$[-50,50]$	0
Rotated Ackley	$F_{10}(X)=20+e-20\exp(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D}\frac{y_i^{\ 2}}{D}})$ $-\exp(\frac{1}{D}\sum_{i=1}^{D}\cos(2\pi y_i)/D)$ $where\ Y=MX, M\ is\ an\ orthogonal\ matrix$	$[-32,32]$	0
Rotated Schwefel	$F_{11}(X)=418.9829D+\sum_{i=1}^{D}-z_i x_i\sin(\sqrt{\lvert x_i\rvert})$ $where\ z_i=\begin{cases}y_i\sin(\sqrt{\lvert y_i\rvert}),\ if\ \lvert y_i\rvert\le 500\\ 0, \quad otherwise\end{cases}, y_i=y'_i+420.96,$ $where\ Y'=M(X-420.96),\ M\ is\ an\ orthogonal\ matrix$	$[-500,500]$	0
Rotated Rastrigin	$F_{12}(X)=\sum_{i=1}^{D}[y_i^{\ 2}-10\cos(2\pi y_i)+10]$ $where\ Y=MX, M\ is\ an\ orthogonal\ matrix$	$[-5.12,5.12]$	0
Shifted Sphere	$F_{13}(x)=\sum_{i=1}^{D}(x_i-(i-0.5D))^2$	$[-100,100]$	0
Shifted Rosenbrock	$F_{14}(x)=\sum_{i=1}^{D-1}[100(x_{i+1}-x_i^{\ 2})^2+(x_i-1)^2]+390$	$[-100]$	390
Shifted Rastrigin	$F_{15}(x)=\sum_{i=1}^{D}((x_i-\frac{i}{D})^2-10\cos(2\pi(x_i-\frac{i}{D}))+10)$	$[-5.12,5.12]$	0

TABLE II
EXPERIMENT RESULTS BY DJ-QPSO USING DIFFERENT VALUES FOR C_0

Functions	Statistical characteristics	$C_0 = 10$	$C_0 = 20$	$C_0 = 50$	$C_0 = 100$	$C_0 = 200$
F_1	Mean	**0.00e+00**	**0.00e+00**	**0.00e+00**	**0.00e+00**	**0.00e+00**
	rank	**1**	**1**	**1**	**1**	**1**
F_2	Mean	**9.8950e–33**	1.2664e–27	2.1253e–31	1.3256e–14	6.7769e–09
	rank	**1**	3	2	4	5
F_3	Mean	3.6434e–47	6.3528e–52	**8.2670e–56**	2.8287e–38	1.4979e–27
	rank	3	2	**1**	4	5
F_4	Mean	9.9499e+00	1.5293e–01	**1.1617e–02**	3.0809e+02	3.3926e+02
	rank	3	2	**1**	4	5
F_5	Mean	1.7811e–−11	1.5919e−10	**2.9962e−14**	4.6619e−07	3.7809e−09
	rank	2	3	**1**	5	4
F_6	Mean	2.0489e−01	1.7236e−02	**6.7756e−03**	4.6707e−02	2.7093e−01
	rank	4	2	**1**	3	5
F_7	Mean	1.0009e−11	5.0718e−12	**7.3065e−15**	3.7388e−07	2.1535e−04
	rank	3	2	**1**	4	5
F_8	Mean	3.1796e−03	**2.6162e−03**	5.9370e−03	4.6099e−03	6.1722e−02
	rank	2	**1**	4	3	5
F_9	Mean	5.3975e−17	6.7708e−23	**1.3497e−32**	2.3660e−10	5.3832e−15
	rank	3	2	**1**	5	4
F_{10}	Mean	**1.0686e−04**	4.5056e−03	7.4536e−04	6.3683e−03	2.1106e−03
	rank	**1**	4	2	5	3
F_{11}	Mean	2.3349e+03	3.0042e+03	4.1790e+03	5.5886e+04	**1.6489e+03**
	rank	2	3	4	5	**1**
F_{12}	Mean	4.9747e+00	1.4924e+01	**2.9848e+00**	3.2833e+00	6.0701e+01
	rank	3	4	**1**	2	5
F_{13}	Mean	**0.00e+00**	**0.00e+00**	**0.00e+00**	**0.00e+00**	**0.00e+00**
	rank	**1**	**1**	**1**	**1**	**1**
F_{14}	Mean	1.1717e+02	2.3288e+03	4.5271e+01	3.4565e+03	**3.9356e+02**
	rank	5	4	3	2	**1**
F_{15}	Mean	1.6914e+01	7.8580e+02	**7.8614e+00**	3.1527e+03	5.3930e+03
	rank	2	3	**1**	4	5
	Total rank	36	37	**25**	52	55
	Final rank	2	3	**1**	4	5

This paper determines the value of the shared period C_0 by means of experimental verification because no mature theory can prove the scientificity of the value of the iteration interval. DJ-QPSO is tested under the condition of Dim = 30 and T = 1000 iterations, and DJ-QPSO algorithm runs independently 30 times on each test function. Contraction factor σ = 0.01 is selected to ensure the reliability of the experiment.

The experimental results of the DJ-QPSO algorithm using different values of C_0 are shown in Table II.

The Friedman test is used to evaluate the influence of the sharing period C_0. According to the calculation, the chi-squared approximation with F-score = 33.662 (four degrees of freedom) yielded P = 0.000. Therefore, when the significance level P = 0.05, the choice of C_0 has a significant difference.

When the sharing period is C_0 = 50, the DJ-QPSO algorithm ranks the highest. 10 of the 15 benchmark functions of the DJ-QPSO algorithm achieve the optimal accuracy, and the optimization performance is better than other values. The experimental results show that the optimal performance of the DJ-QPSO algorithm is the best when the sharing period is C_0 = 50.

The value of σ in the convergence formula also adopts the experimental method. In theory, the smaller the value of σ, the better the final search, but an extremely small value of σ leads to a longer search time. Thus, selecting the appropriate value of σ is important. The experimental conditions remain unchanged. DJ-QPSO are tested under the condition of Dim = 30 and T = 1000 iterations, and the DJ-QPSO algorithm runs independently 30 times on each test function.

Experiment results by DJ-QPSO using different values for σ are shown in Table III.

TABLE III
EXPERIMENT RESULTS BY DJ-QPSO USING DIFFERENT VALUES FOR σ

Functions	Statistical characteristics	$\sigma = 0.001$	$\sigma = 0.01$	$\sigma = 0.1$	$\sigma = 0.5$
F_1	Mean	**0.00e+00**	**0.00e+00**	1.4619e−298	6.8845e−85
	rank	**1**	**1**	2	3
F_2	Mean	**2.0405e−34**	2.4597e−31	5.8708e−20	1.5701e−10
	rank	**1**	2	3	4
F_3	Mean	2.6161e−43	**7.8682e−56**	4.2633e−26	5.3279e−17
	rank	2	**1**	3	4
F_4	Mean	2.3891e−02	**2.0652e−03**	5.5506e+01	9.4118e+03
	rank	2	**1**	3	4
F_5	Mean	1.0066e−10	**2.1889e−13**	1.0944e−06	8.9546e−04
	rank	2	**1**	3	4
F_6	Mean	**3.4413e−04**	4.4282e−03	4.2972e−01	1.2167e+00
	rank	**1**	2	3	4
F_7	Mean	7.3919e−12	**4.4408e−14**	1.4427e−06	8.4035e−05
	rank	2	**1**	3	4
F_8	Mean	2.4622e−01	**4.0044e−03**	3.1958e+03	5.1603e+03
	rank	2	**1**	3	4
F_9	Mean	5.7786e−19	**1.0896e−25**	2.3614e−08	3.6301e−08
	rank	2	**1**	3	4
F_{10}	Mean	2.2054e−03	**4.6510e−04**	1.5410e−01	1.1514e+00
	rank	2	**1**	3	4
F_{11}	Mean	**1.5308e+03**	3.4592e+03	4.0399e+05	7.6715e+06
	rank	**1**	2	3	4
F_{12}	Mean	**2.5869e+01**	1.1939e+02	2.1179e+04	3.7811e+05
	rank	**1**	2	3	4
F_{13}	Mean	1.1297e−309	**0.00e+00**	3.4088e−146	2.2727e−87
	rank	2	**1**	3	4
F_{14}	Mean	2.7397e+02	**3.2965e+02**	9.0455e+05	4.5085e+07
	rank	2	**1**	3	4
F_{15}	Mean	6.9647e+00	**1.2934e+00**	2.8527e+04	8.3430e+06
	rank	2	**1**	3	4
	Total rank	25	**19**	44	59
	Final rank	**2**	**1**	**3**	**4**

The preceding results are also tested by Friedman. According to the calculation, the chi-squared approximation with F-score = 41.230 (three degrees of freedom) yielded P = 0.000. Therefore, when the significance level P = 0.05, the choice of contraction factor has a significant difference.

According to the experimental results in the table, the DJ-QPSO algorithm has the most comprehensive performance when the contraction factor σ is 0.01 because 11 of the 15 benchmark functions obtain the optimal results.

In this study, we compare the proposed DJ-QPSO algorithm with various QPSOs comprising the QPSO [1], MOQPSO [4], LQPSO [5], E-QPSO [3], and CQPSO [9]. The specific parameter settings used in each algorithm are the same as those in the original studies. To ensure a fair comparison, we set the variables used in all the algorithms as follows: swarm population N = 50, Dim = 30, and T = 1000 iterations. Each algorithm runs 50 times independently; the best result, mean, and standard deviation errors are determined based on the top 25 runs. The parameter configurations for all selected algorithms are also given in Table IV, which are based on the suggestions in the corresponding references. Table V provides the numerical statistics of the optimization results of the preceding six algorithms. Fig. 2 provides the convergence curve of each test algorithm.

TABLE IV
ALGORITHMS FOR COMPARISON

Algorithm	Reference	Parameter settings
QPSO	[1]	$\alpha \in [0.6,0.8]$
MOQPSO	[4]	Number of hypercubes: 49 for *F1*–*F5* and 64 for *F6*–*F8* β: 1.0–0.5, d = 10
LQPSO	[5]	$\beta_0 = 1.0$; $\beta_1 = 0.5$; u, r, $rnd \in (0.1)$
E-QPSO	[3]	$R1 = 1.0$, $R2 = 1.8$, $h1/2 = 1.0308$, $h2/2 = 1.8$, $d1 = 0.5155$, $d2 = 0.2851$, $J1 = 19.9975$, $J2 = -6.3571$
CQPSO	[9]	$w_{i,t} \sim u[0.9,1]$, $q = 0.5 + \left(\frac{5}{N} - 0.5\right)\frac{t}{T}$, $\alpha = \alpha_{\max} - (\alpha_{\max} - \alpha_{\min})\frac{t}{T}$
DJ-QPSO	Present	$C_0 = 50$, $\sigma = 0.01$

TABLE V
STATISTICAL RESULTS OF ALGORITHM COMPARISON

Functions	Statistical characteristics	QPSO	MOQPSO	LQPSO	E-QPSO	CQPSO	DJ-QPSO
F_1	Mean	2.8721e−02	2.9936e−16	5.3749e−16	1.5197e−04	1.1716e−11	**0.00e+00**
	Std.	6.6258e−05	1.3827e−31	1.2463e−34	4.5208e−09	1.0520e−23	**0.00e+00**
	Best	2.3250e−02	3.6416e−17	5.2960e−16	1.0443e−04	9.4221e−12	0.00e+00
	Worst	3.4162e−02	5.6229e−16	5.4539e−16	1.9951e−04	1.4009e−11	0.00e+00
F_2	Mean	3.5670e+00	2.1819e−14	2.2969e−15	3.2000e−03	1.1230e−10	**2.5342e−35**
	Std.	9.6200e−02	7.2684e−29	9.4319e−31	4.3953e−06	5.4679e−21	**2.9709e−96**
	Best	3.3476e+00	1.5790e−14	1.6101e−15	1.7000e−03	6.0011e−11	2.5342e−35
	Worst	3.7863e+00	2.7847e−14	2.9836e−15	4.7001e−02	1.6458e−10	2.5342e−35
F_3	Mean	1.8002e−02	1.5870e−24	9.7603e−17	6.2463e−08	1.4639e−16	**5.2976e−57**
	Std.	5.7881e−06	1.6448e−48	1.6542e−34	5.4346e−15	2.3720e−32	**2.7152e−139**
	Best	1.6302e−02	6.8017e−25	8.8508e−17	1.0336e−08	3.7483e−17	5.2976e−57
	Worst	1.9703e−02	2.4939e−24	1.0670e−16	1.1459e−07	2.5529e−16	5.2976e−57
F_4	Mean	5.1741e+02	1.3260e+01	1.2219e+01	6.6010e+01	2.6713e+02	**2.7996e−02**
	Std.	2.1243e+05	2.5396e+00	4.4639e+00	6.4817e+03	6.1622e+04	**8.8853e−04**
	Best	1.9150e+02	1.2134e+01	1.0725e+01	9.0813e+00	9.1599e+01	6.9187e−03
	Worst	8.4332e+02	1.4387e+01	1.3713e+01	1.2293e+02	4.4266e+02	4.9074e−02
F_5	Mean	5.6727e+01	3.0346e+01	6.9653e+00	5.1002e−02	7.1116e+00	**4.2633e−14**
	Std.	3.3048e+03	1.2374e+01	5.7788e−07	1.6972e−05	3.3023e+00	**4.0390e−28**
	Best	1.6078e+01	2.7858e+01	6.9647e+00	4.8101e−02	6.7052e+00	2.8422e−14
	Worst	9.7377e+01	3.2833e+01	6.9658e+00	5.3904e−02	7.5179e+00	5.6843e−14
F_6	Mean	2.9761e−02	1.5702e−02	**5.2180e−15**	5.2314e−01	5.9502e−01	5.3463e−03

	Std.	3.0316e−06	1.0844e−10	**2.8498e−29**	5.4990e−01	6.1900e−05	2.5815e−25
	Best	2.6072e−02	1.4802e−02	1.4433e−15	1.1008e−03	5.8942e−01	7.4000e−03
	Worst	3.3450e−02	1.6602e−02	8.9928e−15	1.0450e+00	6.0052e−01	9.9526e−02
F_7	Mean	7.0650e−01	4.3624e−09	9.1742e−10	1.3500e−02	5.7531e−06	**7.9936e−15**
	Std.	8.1403e−02	1.8507e−17	2.8741e−20	7.8694e−05	3.5242e−12	**0.00e+00**
	Best	5.0480e−01	1.3204e−09	7.9754e−10	7.2003e−03	4.4257e−06	7.9936e−15
	Worst	9.0820e−01	7.4044e−09	1.0373e−09	1.9703e−02	7.0806e−06	7.9936e−15
F_8	Mean	3.5684e+03	3.1493e+03	1.6190e+03	2.8737e+03	3.2963e+03	**2.5455e−04**
	Std.	2.8777e+06	1.8633e+05	2.8100e−07	1.2707e+04	5.0586e+05	**1.6544e−24**
	Best	2.3688e+03	2.8441e+03	1.4243e+03	2.7940e+03	2.7934e+03	2.5455e−04
	Worst	4.7679e+03	3.4545e+03	1.8153e+03	2.9534e+03	3.7992e+03	2.5455e−04
F_9	Mean	4.9514e−02	6.5932e−16	6.0546e−16	1.7799e−04	8.5507e−11	**1.3498e−32**
	Std.	2.1475e−04	2.1424e−31	3.9710e−32	1.0596e−08	6.2310e−21	**0.00e+00**
	Best	3.9105e−02	3.3203e−16	4.6456e−16	1.0520e−04	2.9690e−11	1.3498e−32
	Worst	5.9913e−02	9.8661e−16	7.4637e−16	2.5077e−04	1.4132e−10	1.3498e−32
F_{10}	Mean	1.0519e+01	2.8704e−01	1.6406e−02	1.8132e+01	5.9557e+00	**1.1022e−04**
	Std.	1.4137e+02	2.4548e−04	5.9295e−05	3.8090e−04	1.9930e+00	**0.00e+00**
	Best	2.0582e+00	1.7600e−02	1.0912e−02	1.8118e+01	4.9575e+00	8.7697e−05
	Worst	1.8980e+01	3.9808e−02	2.1800e−02	1.8146e+01	6.9540e+00	1.1882e−04
F_{11}	Mean	1.1163e+04	1.1104e+04	1.4670e+04	1.0931e+04	1.3368e+04	**1.0817e+04**
	Std.	3.3990e−03	3.1287e+03	**9.2995e−08**	2.9361e+04	7.6070e−17	2.2485e−05
	Best	1.1122e+04	1.1065e+04	1.0660e+04	1.0810e+04	1.0863e+04	1.0817e+04
	Worst	1.1204e+04	1.1144e+04	1.6589e+04	1.1053e+04	1.3810e+04	1.0817e+04
F_{12}	Mean	3.0936e+02	1.1591e+02	8.5185e+00	5.9705e+01	3.9576e+02	**2.2732e+00**
	Std.	3.2355e+03	2.1837e+02	8.0597e−04	9.7216e+01	8.6037e+01	**5.5690e−05**
	Best	2.6914e+02	1.0546e+02	8.4984e+00	5.2733e+01	3.8920e+02	2.2679e+00
	Worst	3.4958e+02	1.2636e+02	8.5385e+00	6.6677e+01	4.0231e+02	2.2785e+00
F_{13}	Mean	3.4340e−01	1.6314e−14	6.7799e−16	2.9348e−04	8.4320e−10	**0.00e+00**
	Std.	2.4570e−04	2.7432e−29	1.2467e−31	1.2784e−09	3.4319e−20	**0.00e+00**
	Best	3.3231e−01	1.2611e−14	4.2832e−16	2.6820e−04	7.1220e−10	0.00e+00
	Worst	3.5448e−01	2.0018e−14	9.2766e−16	3.1877e−04	9.7420e−10	0.00e+00
F_{14}	Mean	2.4026e+03	4.0421e+02	3.9105e+02	4.0554e+02	4.0575e+02	**3.9092e+02**

	Std.	1.4482e+06	2.3812e+04	4.6065e−04	3.0138e+01	2.6868e+00	**4.0919e−08**
	Best	1.5517e+03	4.0312e+02	3.9104e+02	4.0165e+02	4.0459e+02	3.9002e+02
	Worst	3.2536e+03	4.0530e+02	3.9107e+02	4.0942e+02	4.0690e+02	3.9183e+02
F_{15}	Mean	1.8047e+02	1.4955e+02	2.3839e+01	3.7156e+01	1.2200e+02	**2.1303e+01**
	Std.	1.7226e+03	4.4879e+02	2.2183e−01	4.5767e+00	1.1734e+03	**1.5314e−01**
	Best	1.5112e+02	1.3457e+02	2.3506e+01	3.5643e+01	9.7781e+01	2.1026e+01
	Worst	2.0982e+02	1.6453e+02	2.4172e+01	3.8669e+01	1.4622e+02	2.1580e+01

TABLE VI
RANKING OF ALGORITHMIC PERFORMANCE

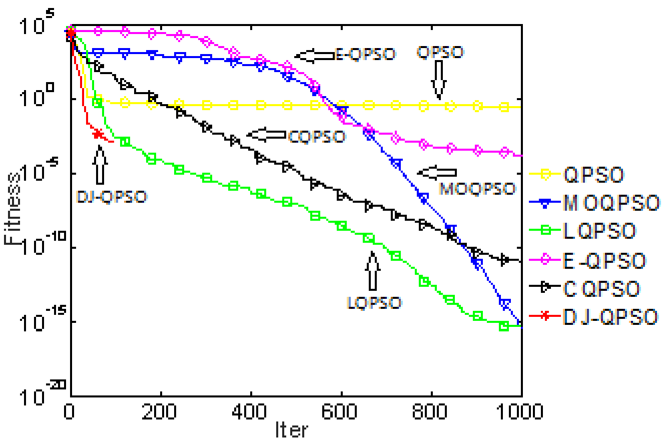
Fun	QPSO	MOQPSO	LQPSO	E-QPSO	CQPSO	DJ-QPSO
F_1	6	2	3	5	4	**1**
F_2	6	3	2	5	4	**1**
F_3	6	2	3	5	4	**1**
F_4	6	3	2	4	5	**1**
F_5	4	3	5	2	6	**1**
F_6	3	4	**1**	5	6	2
F_7	6	3	2	5	4	**1**
F_8	6	4	2	3	5	**1**
F_9	6	3	2	5	4	**1**
F_{10}	4	3	2	5	6	**1**
F_{11}	4	3	6	**1**	5	2
F_{12}	5	4	2	3	6	**1**
F_{13}	6	3	2	5	4	**1**
F_{14}	6	3	2	4	5	**1**
F_{15}	6	4	2	3	5	**1**
Total rank	80	47	38	60	73	**17**
Final rank	6	3	2	4	5	**1**

According to the experimental data in Table V, when the unimodal function F1–F4 is optimized, the optimization results of various improved QPSOs are excellent, among which the DJ-QPSO algorithm has the highest precision and the optimized results are obviously better than other algorithms. When the algorithm optimizes function F1:*sphere*, the optimal result reaches the ideal global optimal value, and the DJ-QPSO shows higher optimization precision and better algorithm execution ability. According to the optimization result of ill-conditioned function F4, the optimization result of the improved algorithm DJ-QPSO is not ideal. At the same time, we can see that other improved algorithms cannot effectively optimize this complex unimodal function, but the accuracy of the DJ-QPSO algorithm is still more accurate in many comparative optimization algorithms and has relatively better performance. The results show that the DJ-QPSO algorithm has room for improvement in optimizing this type of functions. For the multimodal function F5, four optimization algorithms exist, and the optimization effect is not ideal. The DJ-QPSO algorithm can optimize this function with many local minimum values, indicating that the algorithm has good global search performance. For multimodal function F6, the DJ-QPSO algorithm does not show the advantages of the improved algorithm and is inferior to LQPSO in search accuracy. For multimodal function F7, the improved algorithm DJ-QPSO still shows good optimization ability. For multimodal functions F8 and F9, the optimization precision of the DJ-QPSO algorithm is higher than that of other algorithms, and the mean and standard deviation errors of the algorithm are the smallest. The optimization performance of the DJ-QPSO algorithm is the best among these functions. When the rotation function F10 of the unimodal function F7 is optimized, the performance of several improved algorithms decreases due to the increased complexity of the function, but the optimization ability of the

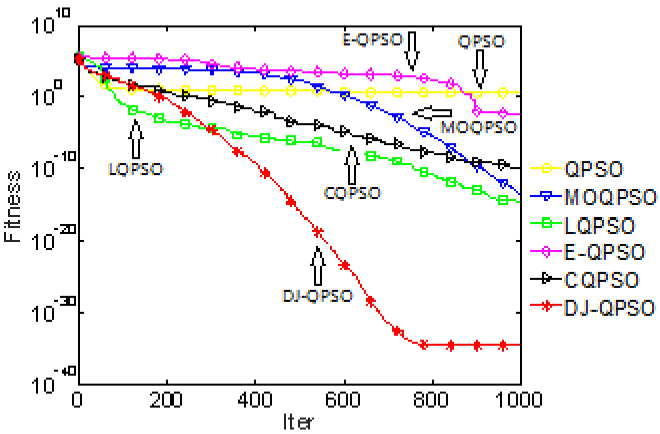
DJ-QPSO algorithm is still the best. The rotation function F12 compares with the function F5, which also indicates that the search precision of the improved algorithm decreases, but the DJ-QPSO search results are still the best. The performance of the DJ-QPSO algorithm in the offset function is still excellent although the search accuracy has decreased. Combined with the optimization results of the DJ-QPSO algorithm in 15 benchmark functions, the performance of the DJ-QPSO algorithm in unimodal function is outstanding. Although the performance of some complex multimodal functions is slightly inadequate, the average optimization performance is still better than that of several other improved algorithms, and the optimization effect is better. These results prove that the improvement strategy of the DJ-QPSO algorithm has an obvious effect.

Table VI shows the performance of the ranking algorithm and provides an overall comparison among all the algorithms. It can be observed that DJ-QPSO has the best total rank, which means it has the best overall performance of all test functions. we can obtain the order: DJ-QPSO, LQPSO, MOQPSO, E-QPSO, CQPSO, QPSO.

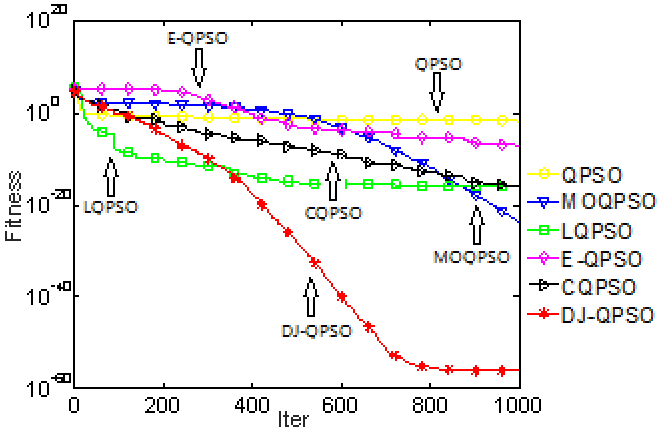
Fig. 2 shows the convergence curves of six improved algorithms, which can indicate the convergence effect of the algorithm intuitively and further verify the experimental results.



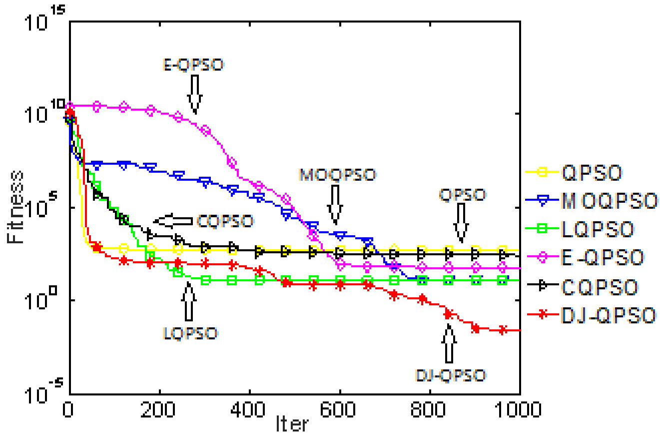
（a） F_1



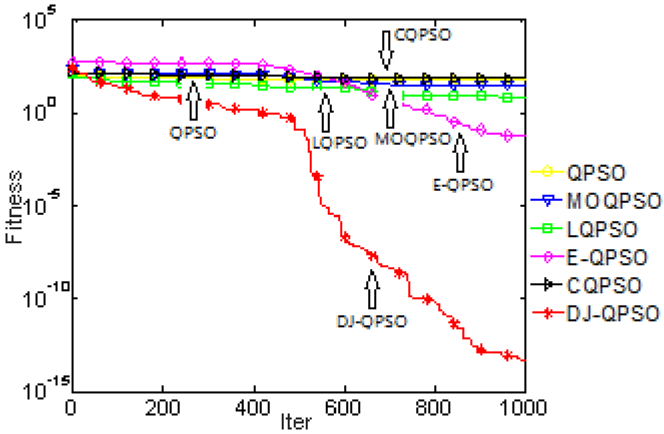
（b） F_2



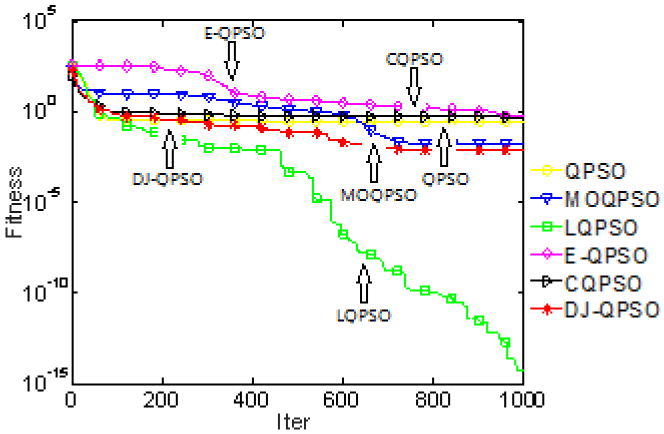
（c） F_3



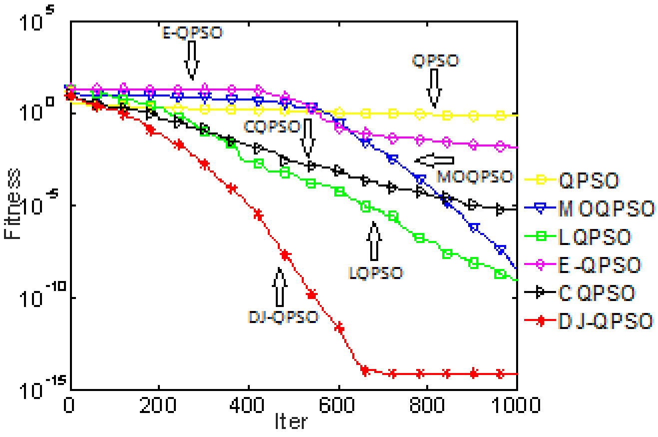
（d） F_4



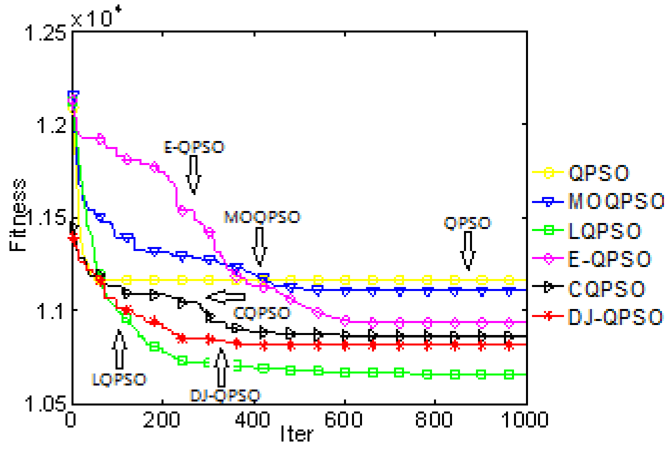
（e） F_5



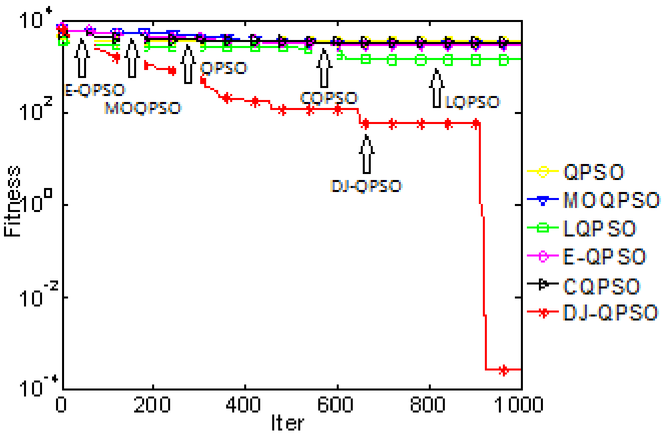
（f） F_6



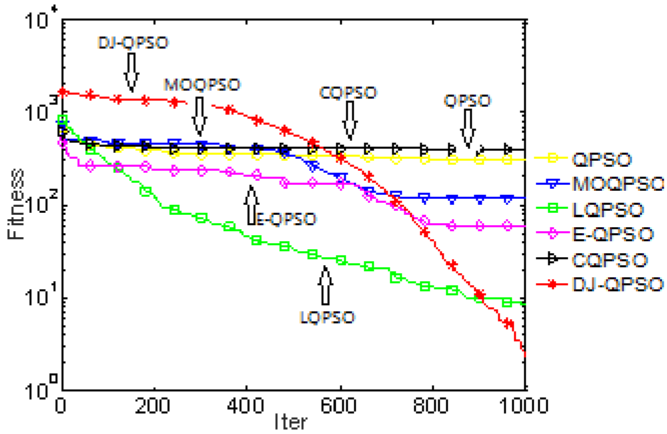
（g） F_7



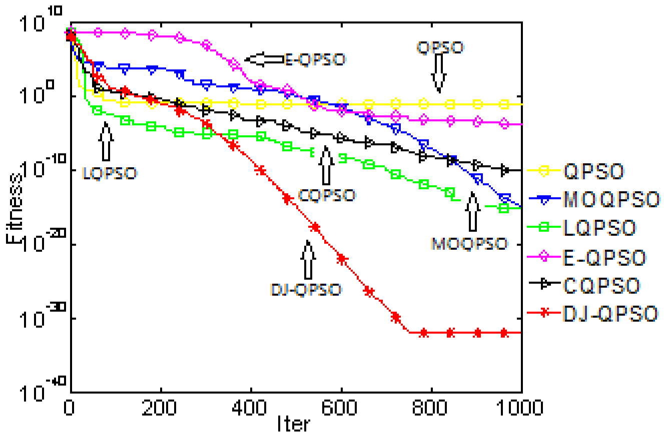
（k） F_{11}



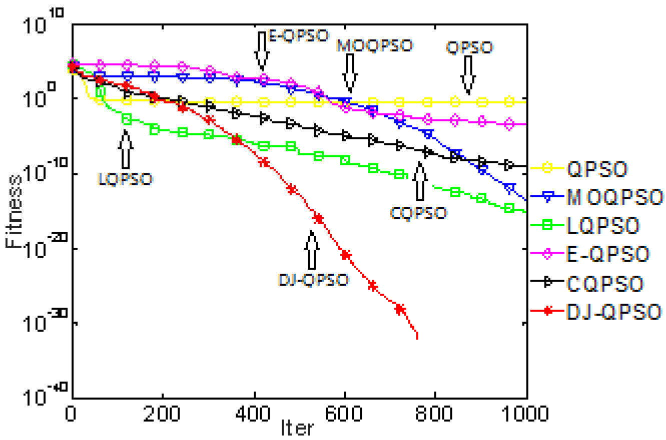
（h） F_8



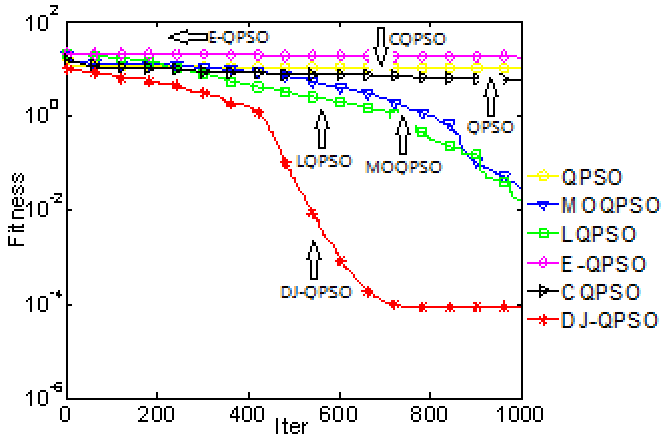
（l） F_{12}



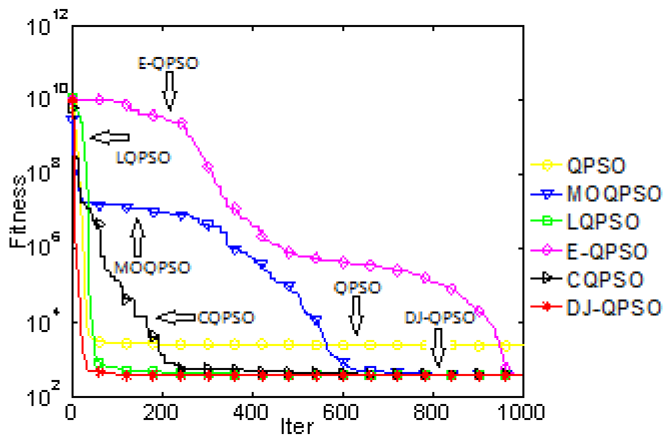
（i） F_9



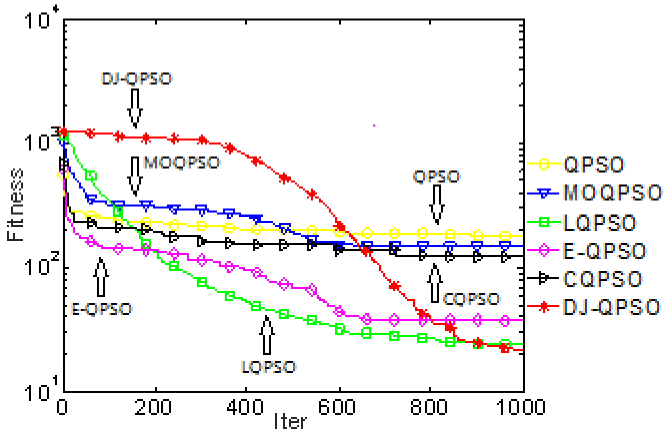
（m） F_{13}



（j） F_{10}



（n） F_{14}



(o) F_{15}

Fig. 2 Contrastive function convergence curve

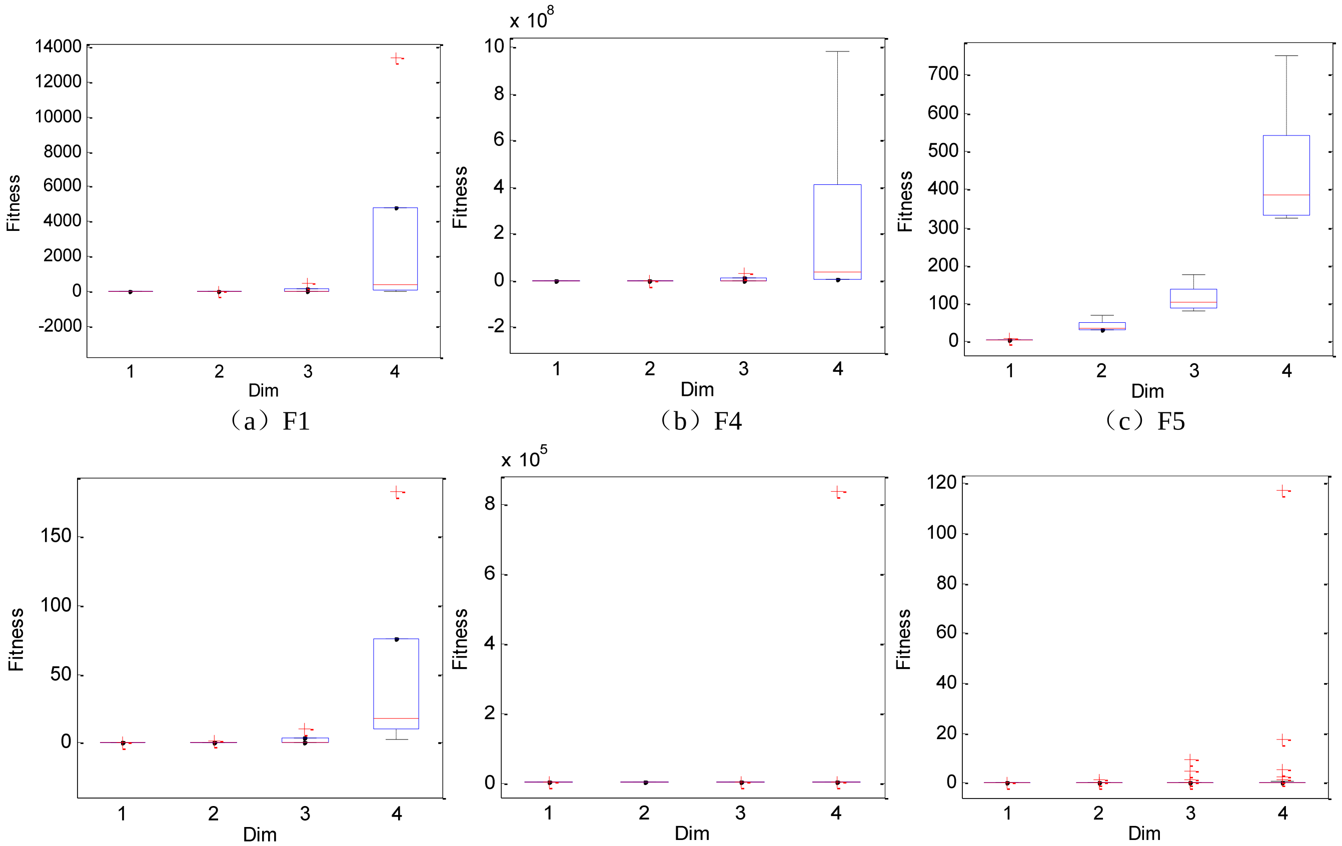
Fig. 2 shows the convergence curves obtained using the various QPSOs for 15 benchmarks, which clearly exhibit the optimization of the algorithm. First, for function F1, several improved algorithms optimize the QPSO to a certain extent. However, the DJ-QPSO algorithm converges fastest, and the global optimal value is found when the number of iterations is approximately 100. The algorithm has advantages in convergence speed and precision. For unimodal functions F2 and F3, the convergence curves of DJ-QPSO are fast in the early stage and slow in the later stage of the algorithm, because in the early stage, the two subgroups quickly search and locate the optimal solution region in the global scope, and the algorithm finely searches around the optimal solution in the later stage. Thus, the algorithm curve presents the convergence trend in Fig. 2 (b)–(c). According to the optimization curve of unimodal function F4, the convergence of the DJ-QPSO algorithm is extremely slow when the number of iterations is approximately 50, and it is likely to fall into the local optimal value, leading to the convergence failure of the algorithm. When the multimodal function F5 is optimized, the convergence curve has an obvious turning point, which indicates that the DJ-QPSO algorithm particles jump out of the local optimal value and find an improved value. When the multimodal function F6 is optimized, the DJ-QPSO algorithm does not show good optimization performance, while the LQPSO algorithm exhibits a better optimization effect and higher accuracy. The convergence curves of the multimodal functions F7 have the same trend as those of unimodal functions F2 and F3. When the multimodal function F8 is optimized, the convergence curve shows that the algorithm can help the particles to jump out of the local optimal value. When the improved algorithm DJ-QPSO optimizes rotation functions F10, F11, and F12, the optimization accuracy of the algorithm decreases significantly. When the algorithm optimizes function F11, DJ-QPSO converges faster at the early stage of optimization and slowly at the later stage of optimization, which fails to achieve the desired optimization effect. It shows that the optimization ability of the algorithm needs to be improved. When the rotation function F12 is optimized, the other five comparison algorithms converge faster than DJ-QPSO in the early stage of optimization, but after 550 generations, DJ-QPSO converges faster than other comparison algorithms, after 900 generations, DJ-QPSO converges faster than all comparison algorithms. Observing the convergence curve of migration functions F14 and F15, we can find that with the increase of the complexity of the function, the algorithm is more likely to fall into local optimum, which affects the performance of the algorithm [24].

Based on the preceding analysis, although the improved DJ-QPSO algorithm performs slightly worse in the optimization of individual multimodal functions, the comprehensive performance is better than that of other comparative algorithms.

To further analyze the optimization performance of the DJ-QPSO algorithm, we use the improved algorithm to optimize unimodal functions F1 and F4, multimodal functions F5 and F6 of benchmark functions with dimensions of 10, 30, 50, and 100. The number of iterations is 1,000. The average results of 25 independent optimizations are counted and analyzed by boxplot method.

A boxplot is a statistical chart used to display a set of data scattering. This method can show the maximum, minimum, median, upper and lower quartiles, and outliers of a set of data. In the figure, half of the data are distributed inside and outside the box, the median corresponding to the horizontal line in the box, and half are distributed up and down. The top side of the box represents the upper four scores, and the upper quarter of the data is distributed; the lower side of the box represents the lower four scores, and the lower quarter of the data is distributed under it. The line connecting the top and bottom of the box is called the tentacle line. The small horizontal line connecting the upper tentacle line represents the maximum value of the statistical data, and the small horizontal line connecting the lower tentacle line represents the minimum value of the statistical data. The "+" in the figure represents singularity and is an outlier in the statistical data. Therefore, the shorter the box, the shorter the tentacles and the fewer singularities, the more centralized the statistical data; on the contrary, the more scattered the statistical data [25].

Fig. 3 shows the statistical result of the DJ-QPSO algorithm to optimize the benchmark function in different dimensions. The abscissa in the boxplot represents the dimension, and the four columns correspond to the optimization results of 10, 30, 50, and 100 dimensions. Fig. 3 (a) is a unimodal function F1. In the optimization results in 10-dimensional space, the boxplot shrinks to a horizontal line and its fitness value is 0. This result shows that in 10-dimensional space, the results obtained by the DJ-QPSO algorithm optimizing sphere function 25 times independently converge to the optimal value, which shows that the algorithm has excellent convergence. In the 30-dimensional space, the upper and middle edges of the boxplot coincide basically, which shows that at least half of the optimization results converge to the optimal value in 25 independent operations. The other half of the optimization results converge to the optimal value, and the accuracy of the singularities is also extremely high. In the 50-dimensional space, the height of the box increases slightly, and singularities exist. However, few deviations occur in the statistical data, and the optimization results are slightly poor.



（a） F1 （b） F4 （c） F5

（d） F6 （e） F5 （f） F6

Fig. 3 Statistical boxplot of optimized result

In the 100-dimensional space, the height of the box increases significantly, and small horizontal lines are found at the bottom of the box to indicate the minimum value. Singularities with large differences appear, and the optimization accuracy is low. Fig. 3 (b) shows a unimodal function F4. The results of F4 with the same unimodal function F1 are almost the same. In the limited iteration time, the optimization accuracy of DJ-QPSO decreases with the increase of dimension.

Fig. 3 (c)–(d) shows the statistical results of optimizing multimodal functions F5 and F6. The boxplot increases with the increase of dimension space, which shows that the statistical data are increasingly scattered, indicating that the optimization accuracy of the algorithm decreases with the increase of dimension. Fig. 3 (e)–(f) shows the optimization result of multimodal functions F5 and F6 when the number of iterations is 2000. We can see intuitively that the number of iterations increases and the statistical data are concentrated. This condition shows that the optimization accuracy of the statistical results in high-dimensional space can be improved by increasing the optimization time of the algorithm.

The dual-population search strategy improves the precision of the algorithm but also requires increased optimization time. How to obtain higher search precision at a lower time cost will be determined based on the key points to be studied later[26]–[28].

V. CONCLUSION

In the DJ-QPSO algorithm, the optimization performance is improved by setting two attraction points and two subgroups. The increase in population diversity adjusts the exploration and exploitation abilities of the algorithm to help the particles escape from the local optimal value. Periodic dynamic-sharing strategy can improve the convergence ability of the population. The global convergence formula is introduced to improve the search precision. The simulation results of benchmark functions indicate that the DJ-QPSO algorithm is able to enhance diversity and optimize the global situation. Further, statistical analysis of experimental results shows that DJ-QPSO algorithm has better comprehensive performance than other related algorithms in solving complex function optimization problems. It is necessary to solve optimization problems in industrial, social, economic, management and other fields. QPSO algorithm is a general optimization algorithm, DJ-QPSO has better optimization performance. It has good application prospects and can be effectively applied in scientific research and engineering fields.

ACKNOWLEDGMENT

This work is supported by the National Science Foundation of China under Grant No. 61473054. The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers to improve the presentation.

REFERENCES

[1] SUN Jun. Quantum behaved particle swarm optimization algorithm[D]. Control theory and control engineering,2009.
[2] Li Y, Xiang R, Jiao L, et al. An improved cooperative quantum-behaved particle swarm optimization[J].SOFT COMPUTING, 2012, 16(6):1061-1069.

[3] Dos SCL, Alotto P. Global Optimization of Electromagnetic Devices Using an Exponential Quantum-Behaved Particle Swarm Optimizer[J]. IEEE Transactions on Magnetics, 2008, 44(6):1074-1077.

[4] Liu T, Jiao L, Ma W, et al. A new quantum-behaved particle swarm optimization based on cultural evolution mechanism for multiobjective problems[J]. Applied Soft Computing, 2016, 46(C):267-283.

[5] Turgut O E,Turgut M S, Coban M T. Chaotic quantum behaved particle swarm optimization algorithm for solving nonlinear system of equations[J].Computers & Mathematics with Applications, 2014,68(4):508-530.

[6] HUANG Ze-xia, YUYou-hong,HUANG De-cai.Quantum-Behaved Particle Swarm Algorithm with Self-adapting Adjustment of Inertia Weight[J].Journal of Shanghai Jiaotong University,2012, V46(2):228-232.

[7] WANG Xiaogen, LONG Hai-xia, SUN Jun. Quantum-behaved particle swarm optimization based on Gaussian disturbance[J]. Application Research of Computer,2010, 27(6):2093-2096.

[8] WU Tao, YAN Yu-song, CHEN Xi. Improved dual-group interaction QPSO algorithm based on random evaluation[J].Control and Decision,2015(3):526-530.

[9] Chen Hanwu, Zhu Jianfeng Ruan Yue. Quantum particle swarm optimization algorithm with crossover operator[J].Journal of Southeast University (Natural Science Edition),2016, 46(1):23-29.

[10] ZHANG Guo-yong, WU Yong-gang and GU Wei. Quantum-behaved particle swarm optimization algorithm based on elitist learning[J]. Control and Decision, 2013, 28(9):1341-1348.

[11] CHEN Wei, ZHOU Di, XU Weibo. Improved quantum-behaved particle swarm optimization algorithm based on comprehensive learning strategy[J].Control and Decision. 2012, 27(5):719-723.

[12] JIA Dong-li, ZHANG Jia-shu. Niche particle swarm optimization combined with chaotic mutation[J].Control and Decision.2007, 22(1):117-120.

[13] WANG Ying, LI Pan-chi. Dual population quantum particle swarm optimization based on chaotic optimization[J].Information Technology,2013(8):89-91.

[14] Chen W, Fu Y, Sun J et al. Improved binary quantum-behaved particle swarm optimization clustering algorithm[J].Control & Decision, 2011, 26(10):1463-1468.

[15] Xi M L, Sun J, Wu Y. Quantum-behaved particle swarm optimization with binary encoding[J]. Control & Decision, 2010, 25(1):376-385.

[16] Yuan S, Shen X and Shao N. Realization of improved Quantum Particle Swarm Optimization algorithm based on search operator of artificial bee colony[J].Computer Engineering and Applications, 2016.52(15):29-33

[17] WANG Yuan-yuan, ZENG Jian-chao and TAN Ying. Cooperative evolutionary particle swarm optimization algorithm with multi-populations[J].Computer Engineering and Design,2007, 28(15):3661-3664.

[18] Zhipeng Li, Weizhong Li and Jiang Y, et al. Quantum-Behaved Particle Swarm Optimization Algorithm Using Niche and Opposition-Based Learning[J].Journal of Chongqing University of Technology, 2018.(1):181-187

[19] Cheng R, Jin Y. A social learning particle swarm optimization algorithm for scalable optimization[J].Information Sciences,2015, 291(6):43-60.

[20] Liang J J, Qin A KS, uganthan P N , et al. Comprehensive learning particle swarm optimizer for global optimization of multimodal functions[J].IEEE Transactions on Evolutionary Computation, 2006, 10(3):0-295.

[21] Han M , Fan J and Wang J. A Dynamic Feedforward Neural Network Based on Gaussian Particle Swarm Optimization and its Application for Predictive Control[J].IEEE Transactions on Neural Networks, 2011, 22(9):1457-1468.

[22] Qi W. A hybrid-forecasting model based on Gaussian support vector machine and chaotic particle swarm optimization[J].Expert Systems with Applications, 2010, 37(3):2388-2394.

[23] HX Long, H Fu and C Shi. Quantum-Behaved Particle Swarm Optimization Based on Diversity-Controlled[J].Ifip Advances in Information & Communication Technology,2016, 445:132-143.

[24] J Wang, LI Zheng and YE Wen. Multi-plane Cooperation Task Assignment Problem Based on Quantum-behaved Particle Swarm Optimization Algorithm[J].Ship Electronic Engineering, 2017,8:28-33.

[25] Wang Z. A Chaos Quantum-behaved Particle Swarm Optimization Algorithm Based on Differential Evolution[J].Computer & Modernization, 2017.08:22-30.

[26] Jie-Sheng Wang, and Shu-Xia Li, "PID Decoupling Controller Design for Electroslag Remelting Process Using Cuckoo Search Algorithm with Self-tuning Dynamic Searching Mechanism," Engineering Letters, vol. 25, no.2, pp125-133, 2017

[27] Yuanyuan Pan, and Yongdong Shi, "A Grey Neural Network Model Optimized by Fruit Fly Optimization Algorithm for Short-term Traffic Forecasting," Engineering Letters, vol. 25, no.2, pp198-204, 2017

[28] ie-Sheng Wang, and Jiang-Di Song, "A Hybrid Algorithm Based on Gravitational Search and Particle Swarm Optimization Algorithm to Solve Function Optimization Problems," Engineering Letters, vol. 25, no.1, pp22-29, 2017

Jinling Niu is currently pursuing an M.S. degree in Control Science and Engineering from the School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, China.

Yong Zhang. received B.S. and M.S. degrees in Control Science and Engineering from the University of Science and Technology Liaoning, Anshan, in 1987and 1990, and a Ph.D. in control science from the Dalian University of Technology, China, in 2005. He is currently a Professor and a graduate student supervisor in the School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, China. His main research area of interest is the modelling of complex industry processes and intelligent control.