Crossover Operation of Random Drift Particle Swarm Optimization

1st Min Yan
Jiangnan University
Wuxi, China
1071841664@qq.com

2nd Jun Sun
Jiangnan University
Wuxi, China
sunjun_wx@hotmail.com

3rd Qidong Chen
Jiangnan University
Wuxi, China
cqd_jnu@hotmail.com

***Abstract*—In order to enhance the global search capability, a crossover operation based on RDPSO algorithm(CO-RDPSO) is put forward. Through crossover operation, particles can skip the local optimum and strengthen the global search capability. Test CO-RDPSO with twelve functions, the experiment results directions, CO-RDPSO not only obtains better results but also converge faster than other algorithms both on unimodal or multimodal.**

Keywords—random drift particle swarm, crossover operation, converge

I. INTRODUCTION

Most mechanical design, optimal path, control engineering, and other issues can be considered optimization problems. Therefore, many optimization algorithms have appeared. PSO is one of the optimization algorithms.

PSO[1] has been applied to various fields by scholars. However, PSO is prone to stuck in a local optimum. Researchers have proposed many solutions to this problem. Mendes, R et al. Proposed in 2004 that FIPSO[2].Although FIPSO has the characteristics of fast convergence, it still misses outstanding areas. Sun J proposed the Random Drift PSO (RDPSO) algorithm in 2015. The main disadvantage is that particles at high latitudes, no optimal global solution can be found.

In order to mitigate the problem of particles are trapped to local optimum easily, a CO-RDPSO algorithm is proposed. The advantages are:firstly, the optimal historical positions of two individuals are randomly selected to generate new particles, which can let the swarm converge slower.Secondly, after the crossing operation, the new particle is expected to become the guider of the particle swarm so as to guide the trapped particles are out the small area.

Other parts of this article: Section Ⅱ is the related concepts of RDPSO and crossover. Section Ⅲ introduces the principle and pseudocode of CO-RDPSO. Section Ⅳ is the parameter settings, test functions, experimental results, and analysis.The fifth part is the conclusion.

II. BASIC OPERATION

A. RDPSO

Sun J proposed RDPSO, and the RDPSO regards the electro-nic drift motion as the local search for the minimum potential energy[3]. Random movement prevents particles from falling into a local minimum potential energy (locally optimal). Therefore, in the RDPSO algorithm, at the *n+1-th* iteration step, the particle velocity is composed of random motion (thermal motion) and drift motion (directional motion)

$$V^j_{i,n+1}=\alpha\left|C^j_n\text{-}X^j_{i,n}\right|\varphi^j_{i,n+1}+\beta(P^j_{i,n}\text{-}X^j_{i,n}) \tag{1}$$

$X^j_{i,n}$ is the position of the *i-th* particle of the *j-th* component at the *n-th* iteration. $\alpha\left(C^j_n\text{-}X^j_{i,n}\right)\varphi^j_{i,n+1}$ means random motion. $\beta(P^j_{i,n}\text{-}X^j_{i,n})$ means drift motion. C^j_n represents the mean personal best solution. $p^j_{i,n}=\phi^j_{i,n}P^j_{i,n}+(1\text{-}\phi^j_{i,n})G^j_n$. $\phi^j_{i,n}$ is distributed on [0,1]. The formula for the *n+1-th* iteration of RDPSO is:

$$X^j_{i,n+1}=\left(X^j_{i,n}+V^j_{i,n+1}\right) \tag{2}$$

In the RDPSO algorithm. By controlling the values of α and β, it is possible to dynamically adjust global areas and local areas.

B. Crossover operation

Holland[4] proposes Genetic Algorithm(GA). The shortages of GA are easily precocity, and the convergence is slow. To overcome these problems, scholars have proposed crossover operators. Crossover is considered to be one of the most important strategies in genetic algorithms (GA). Chen redefines the $W^j_{i,n}$ formula for crossing operation and proposed PSOCO algorithm, in which randomly select $P^j_{i1,n}$and $P^j_{i2,n}$ is defined as following[5]:

$$W^j_{i,n}=r_1P^j_{i1,n}+(1\text{-}r_1)*P^j_{i2,n} \tag{3}$$

Then, apply (4) to make a judgment：

$$P^j_{i,n}=\begin{cases}W^j_{i,n}, & if\ rand(1)<C\ or\ j=j_{rand}\\ P^j_{i,n}, & esle\end{cases} \tag{4}$$

However, PSOCO can not find a satisfying result in a local optimal state on complex functions.

III. CROSSOVER OPERATION RANDOM DRIFT PARTICLE SWARM OPTIMIZATION

The main disadvantage of this RDPSO is that the leader update direction only exists in the current particle swarm. In this case, all particles cannot be found the best location. Hence, the crossover operation is used to mitigate this disadvantage of RDPSO and proposed a new algorithm CO-RDPSO. CO-RDPSO can not only ease the shortage that the particles fall into the local optimum easily in RDPSO, but also can speed up the convergence speed.

The main steps of CO-RDPSO: First, apply (4) whether to apply (3) to obtain a new particle $W^j_{i,n}$. In (3) both $P^j_{i1,n}$ and $P^j_{i2,n}$ are randomly selected in $P^j_{i,n}$. Otherwise, $P^j_{i,n}$ remains unchanged. Finally, use (1) and (2) to change the speed and position. In terms of time complexity, it is described that CO-RDPSO has the same characteristics as RDPSO is *O(N)*. The pseudo code is shown below.

2473-3636/20/$31.00 ©2020 IEEE
DOI 10.1109/DCABES50732.2020.00071

Pseudo-code of the algorithm：

```
initialize the required parameters
calculate the values of randomly particles
for n=1 to MAX_ITER do
    for i =1 to PS
        for j=1 to D
            construct P_{i,n}^j by using (4) and (3);
            change the velocity and position using (1) and using (2);
            compare fitness;
        update P_{i,n}^j and G_n^j;
    end
end
end
```

IV. EXPERIMENT AND RESULT ANALYSIS

A. *Experimental setup and test functions*

The CO-RDPSO algorithm is compared with PSOCO, ELPSO[6], FIPSO, PSO, QPSO[7], RDPSO, UPSO[8] to test its performance.The swarm size is set to 50, and the maximum number of evaluations equals to $1000D$, D is the dimension.All the experiment results are obtained by conducting 30 times independently. Table 1 shows the settings of the applied algorithm parameters. Table 2 shows the 12 test functions and search spaces.

TABLE I. ARAMETER SETTINGS

Algorithm	**parameters**
PSOCO	W:0.4~0.9,C=1.496181,r=0.05
ELPSO	W:0.4~0.9,C_1=C_2=2
FIPSO	X=0.7298，$\sum C_i$=4.1
PSO	W:0.4~0.9,C_1=C_2=2
QPSO	β:0.5~1
RDPSO	W:0.3~0.9, β=1.45
UPSO	u=0.1,μ=0,x=0.729,σ=0.01
CO-RDPSO	W:0.3~0.9, β=1.45,C=1

TABLE II. TEST FUNCTION

Function		**search range**
F1	Sphere:$f_1(x)=\sum_{i=1}^{D} y^2$,$y=x_i$	$[-100,100]^D$
F2	SchwefelP2.22:$f_2(x)=\sum_1^D\lvert y\rvert+\prod_1^D\lvert y\rvert$, $y=x_i$	$[-10,10]^D$
F3	Rosenbrock:$f_3(x)=\sum_i^{D-1}[100\text{-}y^2)^2+(y-1)^2]$,$y=x_i$	$[-10,10]^D$
F4	Noncontinuous rastrigin:$f_4(x)=\sum_{i=1}^{D}[y^2\text{-}10\cos(2\pi)+10]$,$y=x_i$	$[-5.12,5.12]^D$
F5	Ackley:$f_5(x)=-20\exp\left(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D}y^2}\right)$ - $\exp\left(\frac{1}{D}\sum_{i=1}^{D}\cos 2\pi y\right)+20+e$, $y=x_i$	$[-32,32]^D$
F6	Griewank:$f_6(x)=\frac{1}{4000}\sum_{i=1}^{D}y^2$ - $\prod_{i=1}^{D}\cos\left(\frac{y}{\sqrt{i}}\right)+1$, $y=x_i$	$[0,600]^D$
F7	Rastrigin:$f_7(x)=\sum_{i=1}^{D}[y^2\text{-}10\cos(2\pi y)+10]$, $y=x_i$	$[-0.5,0.5]^D$
F8	Rotated ackley: $f_8(x)=-20\exp\left(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D}y^2}\right)$ -+20+e , y=M*x_i,(M is the rotation matrix)	$-32,32]^D$
F9	Rotated rastrigin:$f_9(x)=\sum_{i=1}^{D}y^2\text{-}10\cos(2\pi y)+10]$, $y=M*x_i$	$[-5.12,5.12]^D$
F10	$Rotated\ weierstrass$: $f_{10}(x)=\sum_{i=1}^{D}(\sum_{k=0}^{kmax}[a_k cos\ (2\pi b_k(y+0.5))]) - D\sum_{k=0}^{kmax}[a_k\ cos(2\pi b_k\times 0.5)]$, $y=M*x_i$	$[-0.5,0.5]^D$
F11	Rotated Elliptic:$f_{11}(x)=\sum_{i=1}^{D}(10^6)^{\frac{i-1}{D-1}}*y^2$, $y=M*x_i$	$[-100,100]^D$
F12	Rotated Noncontinuous rastrigin: $f_{12}(x)=\sum_{i=1}^{D}[y^2-10\cos(2\pi y)+10]$, $y=M*x_i$	$[-5.12,5.12]^D$

B. *Experimental results*

Table 3 shows the performance of CO-RDPSO on 50D optimization problems. The bold value indicates that the effect of this algorithm is higher than other algorithms. To test CO-RDPSO is significantly different from other algorithms; a significance level of 0.05 is used to conduct a two-tailed test. Among them, '+' indicates that CO-RDPSO has a significant advantage over another algorithm; '=' indicates that CO-RDPSO is not significantly different from another algorithm; '-' indicates that algorithm CO-RDPSO has a significant disadvantage with another algorithm.

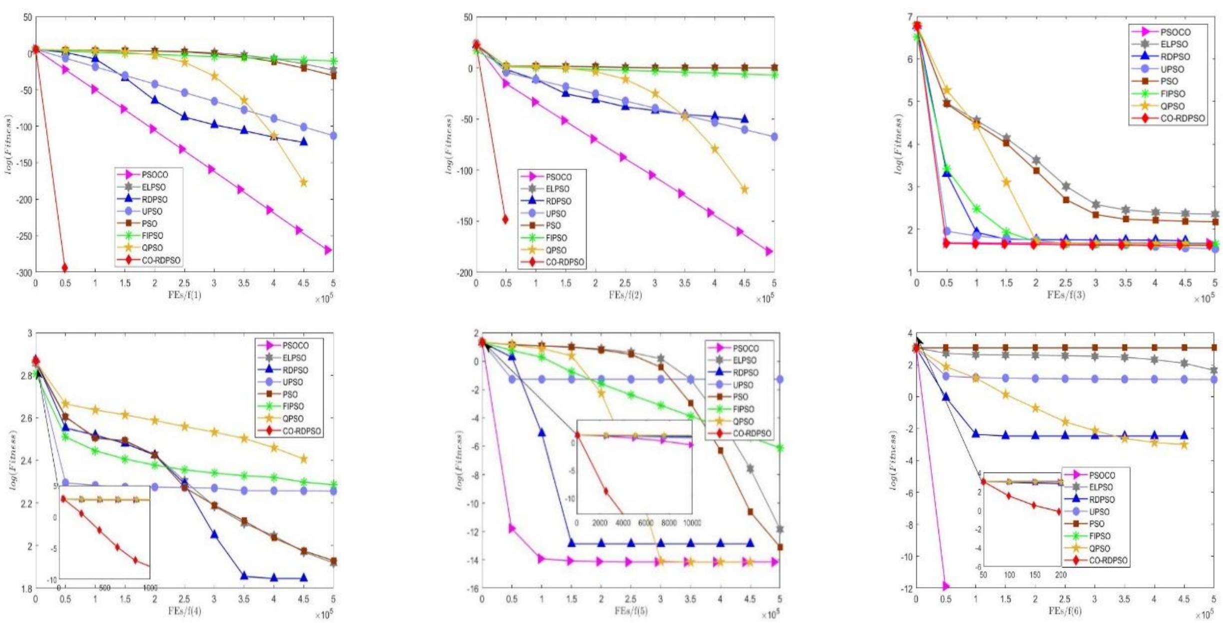
Table 3 shows that for the f_1~f_2, the significant effect of CO-RDPSO is only significantly inferior to that of PSO and RDPSO, but the mean value can find the minimum value of 0. For the f_3, the performance of CO-RDPSO is both mean and significance None of them are good; for multimodal functions f_4~f_7 , CO-RDPSO can quickly find the best solution 0, the significance is not different from PSOCO, but it is better than other algorithms. f_8~f_{12} have an obvious advantage compared with other algorithms, whether they are mean or significant.

Figure 1 describes the convergence effect of each algorithm on 50D optimization problems. The horizontal axis is evaluations size of each algorithm, and the vertical axis display mean value of the particle swarm over 30 times run. Since this algorithm is easy to reach the minimum value of 0, a small graph is used to show the convergence curve of the first few thousand times as the number of evaluations increases. Among f_1 to f_{12}, except for f_3, it can be seen that CO-RDPSO not only converges fast in all functions, but also finds the optimal value of 0. Comprehensive 12 test functions, CO-RDPSO has significant advantages over other algorithms.

From Figure 1 and Table 3, CO-RDPSO can quickly find the optimal solution, and also improves the convergence speed.

TABLE III. DIMENSION D = 50, THE RESULTS OF 12 TEST FUNCTIONS

Function		PSOCO	RDPSO	ELPSO	UPSO	PSO	FIPSO	QPSO	CO-RDPSO
F1	mean	2.87E-270+	8.28E-127-	4.58E-24+	9.21E-114+	1.33E-31-	2.04E-11+	6.62E-252+	**0.00E+00**
	std	0.00E+00	4.54E-126	9.47E-24	1.47E-113	4.45E-31	7.22E-12	0.00E+00	**0.00E+00**
	median	7.95E-273	8.50E-138	9.95E-25	3.22E-114	1.20E-32	1.92E-11	3.45E-254	**0.00E+00**
F2	mean	2.97E-180+	2.70E-57-	1.33E+00+	3.07E-68+	6.67E-01-	8.64E-08+	3.65E-165+	**0.00E+00**
	std	0.00E+00	1.24E-56	3.46E+00	1.75E-68	2.54E+00	1.95E-08	0.00E+00	**0.00E+00**
	median	1.01E-182	1.20E-60	2.56E-18	2.89E-68	1.97E-23	8.48E-08	6.84E-166	**0.00E+00**
F3	mean	4.72E+01+	5.36E+01+	2.27E+02+	**3.47E+01-**	1.49E+02+	4.47E+01+	4.54E+01+	4.17E+01
	std	5.36E-01	2.36E+01	3.43E+02	**2.40E+01**	2.57E+02	5.32E-01	2.56E-01	6.85E-01
	median	4.73E+01	4.19E+01	9.16E+01	**3.11E+01**	8.72E+01	4.48E+01	4.53E+01	4.19E+01
F4	mean	**0.00E+00=**	7.01E+01+	8.29E+01+	1.80E+02+	8.51E+01+	1.93E+02+	2.16E+02+	**0.00E+00**
	std	**0.00E+00**	1.81E+01	3.45E+01	2.83E+01	2.86E+01	2.14E+01	3.83E+01	**0.00E+00**
	median	**0.00E+00**	6.75E+01	8.60E+01	1.86E+02	8.65E+01	1.90E+02	2.17E+02	**0.00E+00**
F5	mean	7.11E-15+	1.34E-13+	1.41E-12+	5.20E-02-	7.96E-14+	7.66E-07+	6.99E-15+	**0.00E+00**
	std	0.00E+00	1.07E-13	1.67E-12	2.85E-01	2.34E-14	1.37E-07	6.49E-16	**0.00E+00**
	median	7.11E-15	1.14E-13	7.53E-13	7.11E-15	7.82E-14	7.34E-07	7.10E-15	**0.00E+00**
F6	mean	**0.00E+00=**	3.40E-03+	4.32E+01+	1.16E+01+	1.12E+03+	**0.00E+00=**	8.49E-04+	**0.00E+00**
	std	**0.00E+00**	5.20E-03	6.72E+00	9.31E-01	7.76E+01	**0.00E+00**	3.30E-03	**0.00E+00**
	median	**0.00E+00**	2.78E-16	4.43E+01	1.13E+01	1.13E+03	**0.00E+00**	0.00E+00	**0.00E+00**
F7	mean	**0.00E+00=**	3.68E+01+	8.68E+01+	1.51E+02+	8.29E+01+	1.93E+02+	6.67E+01+	**0.00E+00**
	std	**0.00E+00**	1.35E+01	2.39E+01	1.68E+01	2.51E+01	1.87E+01	5.84E+01	**0.00E+00**
	median	**0.00E+00**	3.23E+01	8.46E+01	1.53E+02	8.46E+01	1.96E+02	5.92E+01	**0.00E+00**
F8	mean	6.75E-15+	1.28E-13+	2.79E+00+	1.60E+00+	3.00E+00+	1.53E-06+	6.99E-15+	**0.00E+00**
	std	1.08E-15	8.44E-14	5.60E-01	5.23E-01	6.56E-01	3.39E-07	6.49E-16	**0.00E+00**
	median	7.11E-15	1.10E-13	2.77E+00	1.73E+00	2.99E+00	1.50E-06	7.11E-15	**0.00E+00**
F9	mean	1.82E+02+	2.08E+02+	1.68E+02+	1.36E+02+	2.04E+02+	3.56E+02+	3.35E+02+	**0.00E+00**
	std	2.20E+01	9.66E+01	9.45E+01	1.88E+01	1.20E+02	1.67E+01	2.17E+01	**0.00E+00**
	median	1.83E+02	2.08E+02	1.29E+02	1.33E+02	1.24E+02	3.57E+02	3.36E+02	**0.00E+00**
F10	mean	2.27E+01+	9.42E+00+	2.67E+01+	4.54E+01+	2.66E+01+	1.96E+01+	3.12E+01+	**0.00E+00**
	std	8.91E+00	3.27E+00	4.88E+00	3.22E+00	3.95E+00	5.22E+00	8.97E+00	**0.00E+00**
	median	2.00E+01	8.84E+00	2.64E+01	4.63E+01	2.70E+01	1.84E+01	3.36E+01	**0.00E+00**
F11	mean	2.33E-04+	5.30E-03+	2.89E-08-	2.41E-08+	1.95E-10-	3.20E-03+	6.70E-03+	**2.45E-15**
	std	6.29E-04	1.43E-02	1.57E-07	5.37E-08	9.50E-10	5.30E-03	9.00E-03	**1.22E-14**
	median	5.70E-05	8.49E-04	0.00E+00	3.94E-25	0.00E+00	4.91E-04	2.50E-03	**3.01E-108**
F12	mean	2.19E+02+	2.33E+02+	2.31E+02+	2.16E+02+	2.30E+02+	3.22E+02+	3.06E+02+	**0.00E+00**
	std	2.33E+01	8.26E+01	6.39E+01	3.71E+01	7.67E+01	1.86E+01	1.72E+01	**0.00E+00**
	median	2.21E+02	2.15E+02	2.26E+02	2.19E+02	2.23E+02	3.22E+02	3.07E+02	**0.00E+00**



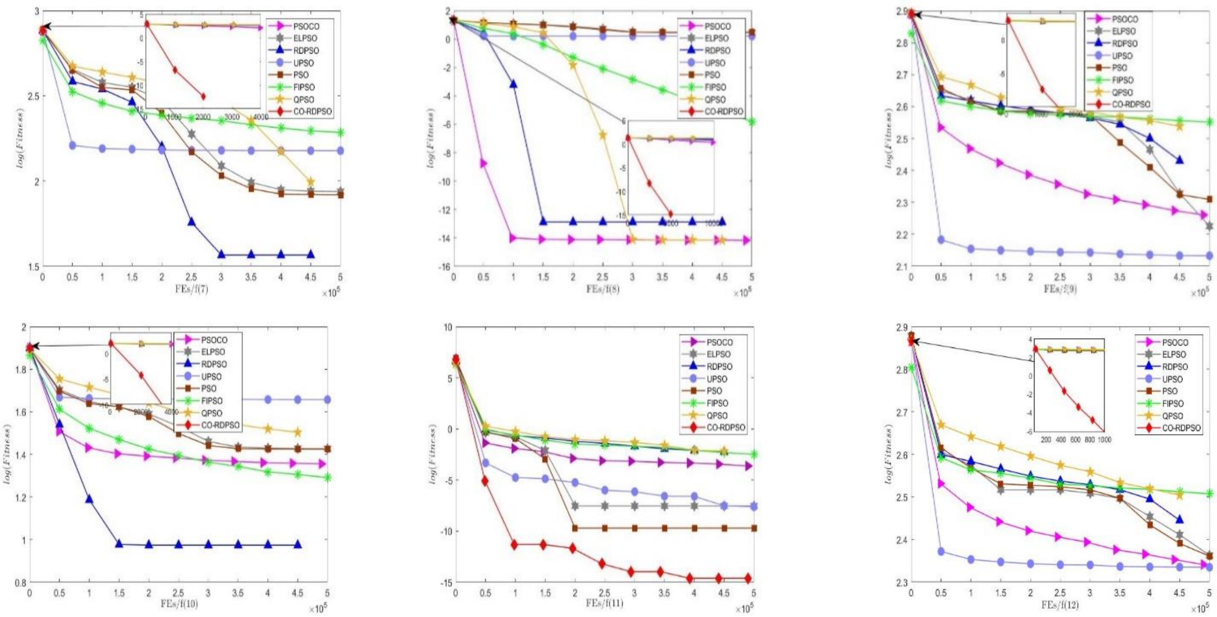


Fig. 1. Convergence curves of each algorithm in 50 dimensions on 12 test functions

C. Parametric research

The most important parameter in CO-RDPSO is C. C determines the position of original $P_{i,n}^{j}$ and the newly constructed particles $W_{i,n}^{j}$. This parameter is critical to the performance of CO-RDPSO. We tested it both on unimodal function and multimodal function. Figure 2 is obtained by conducting 30 times on 10D optimization problems. We can see that when C equals to 1, both the convergence speed and the global best solution it find are better than other parameters. Hence, we set C to 1 in all experiments.

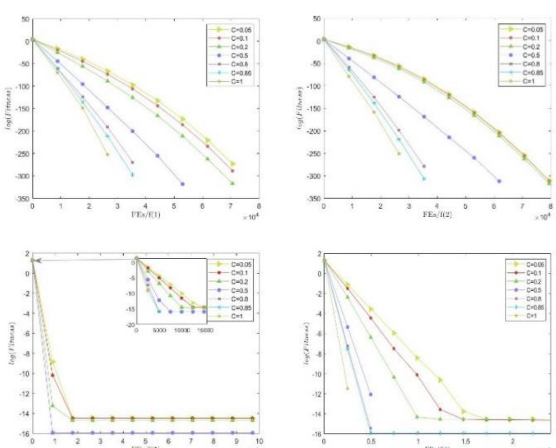


Fig. 2. Convergence curves when C is taken at different values

V. Conclusion

In order to solve the problem of RDPSO falling into local optimality on high-latitude or multi-extreme functions. CO-RDPSO algorithm is proposed. CO-RDPSO integrates the advantages of the crossover operator, which enables the population to add new particles, and at the same time, it is possible for new particles to enter a new update region to find the best location.

References

[1] Eberhart R and Kennedy J. "A new optimizer using particle swarm theory"[C]. Proc Sixth Intl. Symposium on Micro Machine and Human Science,1995: 39-43.

[2] Mendes R, Kennedy J and Neves J. "The fully informed particle swarm: simpler, maybe better"[C]. IEEE Transactions on Evolutionary Computation,2004,8(3):204-210.

[3] Sun J, Wu X,Palade V, Fang W and Shi Y.Random drift particle swarm optimization algorithm: convergence analysis and parameter selection[J]. Machine Learning,2015,101,345-376.

[4] Goldberg D. Genetic Algorithms in Search, Optimization and Machine Learning[C]. Addison-Wesley, New York, 1989.

[5] Chen Y G, Li, Li X, Xiao J H, Yang Y X and Liang J T.Particle swarm optimizer with crossover operation Eng[J]. Appl. Artif . Intell, 2018,70:159-169.

[6] Jordehi A. Enhanced Leader PSO (ELPSO): A new PSO variant for solving global optimisation problems[J]. Applied Soft Computing, 2015, 26:401-417.

[7] Sun Jun,Feng Bin and Xu Wen Bo.Particle swarm optimization with particles having quantum behavior[C]. Proceedings of the IEEE Congress on Evolutionary Computation, Portland, Oregon,2004:325-331.

[8] Parsopoulos K E and Vrahatis M N. "Unified particle swarm optimization for solving constrained engineering optimization problems"[C]. International Conference on Natural Computation, 2005: 582-591.