

PAPER • OPEN ACCESS

## Improved Particle Swarm Optimization Using Wolf Pack Search

To cite this article: Hao-ran Chen *et al* 2019 *J. Phys.: Conf. Ser.* **1176** 052009

View the [article online](#) for updates and enhancements.

You may also like

- [Continuous Wavelet Analysis of Matter Clustering Using the Gaussian-derived Wavelet](#)

Yun Wang, Hua-Yu Yang and Ping He

- [Simultaneous Dependence of Matter Clustering on Scale and Environment](#)

Yun Wang and Ping He

- [Robust simultaneous myoelectric control of multiple degrees of freedom in wrist-hand prostheses by real-time neuromusculoskeletal modeling](#)

Massimo Sartori, Guillaume Durandau, Strahinja Došen et al.



**ECS**  
The  
Electrochemical  
Society  
Advancing solid state &  
electrochemical science & technology

**DISCOVER**  
how sustainability  
intersects with  
electrochemistry & solid  
state science research

# Improved Particle Swarm Optimization Using Wolf Pack Search

Hao-ran Chen<sup>1,\*</sup>, Li-jie Cui<sup>2,a</sup>, Qing Guo<sup>1,b</sup>, Jia-kui Zhang<sup>1,c</sup>

<sup>1</sup>College of Graduate, Air Force Engineering University, Xi'an, China

<sup>2</sup>College of Equipment Management and UAV Engineering, Air Force Engineering University, Xi'an, China

\*Corresponding author e-mail: [1765667894@qq.com](mailto:1765667894@qq.com)

<sup>a</sup>lijie\_cui@163.com, <sup>b</sup>zibai4991@qq.com, <sup>c</sup>794692508@qq.com

**Abstract:** Particle Swarm Optimization (PSO) has excellent global exploration ability, but its local exploitation ability is not ideal. Wolf Pack Search (WPS), which is abstracted from the intelligent predatory behavior of the wolf pack, is an excellent local exploitation strategy and can be used to replace or improve the local exploitation capabilities of other heuristic algorithms. In order to improve the local exploitation ability of Particle Swarm Optimization without affecting the global exploration ability, a hybrid improved algorithm, named as WPS-PSO, based on wolf pack search is proposed. Though the simulation of fixed-dimension and multi-dimensional benchmark functions, and compared with the simulation results of the basic particle swarm algorithm,  $\theta$ -PSO and Quantum Particle Swarm Optimization (QPSO), the results of 50 times simulations show that wolf pack search can improve the local exploit ability of PSO, and can better solve the multi-dimensional optimization problem.

## 1. Overview

Particle swarm optimization algorithm (PSO) is a new type of intelligent group stochastic optimization algorithm based on bird foraging behavior. It is proposed by American electrical engineer Russell Eberhart and social psychologist James Kennedy in 1995[1~3]. The algorithm is a typical heuristic algorithm with the advantages of simple concept, easy implementation, and excellent global search features. However, PSO algorithm performance in local development performance is not ideal enough. In order to optimize the optimization performance of PSO algorithm, many domestic and foreign scholars have done lots of researches and experiments.

In terms of algorithmic analysis, I. C. Trelea[4] analyzed the convergence of the PSO algorithm



applying dynamic system theory and discussed the exploratory and developmental aspects of the PSO algorithm; M. Jiang[5] et al. proved the random convergence of PSO with stochastic process theory, and a lot of experiments were done on the parameter setting of PSO algorithm. In terms of algorithm improvement, scholars have made a lot of optimization of the PSO algorithm on particle selection strategies[6], adaptive inertia weights[7], update formulas[8-10], velocity update strategies[11], and multi-algorithm fusion[12].

In 2007, Yang et al.[13] first proposed a Wolf pack search (WPS) during the process of simulating wolves hunting. WPS is an excellent local development strategy that can be used to replace or improve the local exploitation capabilities of other heuristic algorithms.

In order to improve the local development performance further and fully exploit the excellent global exploration characteristics of the PSO algorithm, this paper combines Wolf Pack Search (WPS) to improve the global version of the particle swarm optimization algorithm and proposes a syncretic algorithm - Particle Swarm Optimization Algorithm Using Wolf Pack Search (WPS-PSO). In addition, a large number of simulations on the basic test function set and high-dimensional test function set were made to verify the effectiveness and advantages of the algorithm in local development performance.

## 2. Basic Particle Swarm Optimization Algorithm

### 2.1 Biological principle of PSO

Particle swarm optimization algorithm (PSO), which simulates the foraging behavior of birds, considers each bird as massless and voluminous particle, the birds' flight space as the solution space of the problem, each passing position of the bird as a candidate solution. Therefore the search for food in a flock is the search for the optimal value. The basic idea of PSO algorithm is to find the global optimal value through the individuals and collaboration and information sharing between individuals. In the algorithm, each solution of the optimization problem is abstracted into particles (birds) without mass and volume, and extends to D dimension. The position and velocity of particle  $i$  in the D dimension are both D dimensional vectors. At the beginning of the algorithm, the initial position  $X_i = (x_{i1}, x_{i2}, \dots, x_{iD})$  of the particles are randomly initialized in the feasible solution space and the initial velocity  $V_i = (v_{i1}, v_{i2}, \dots, v_{iD})$  in the velocity space. In each iteration, the objective function value  $Y_i = f(X_i)$  of each particle was calculated, and the optimal position  $P_i^{best}$  of each particle and the optimal position  $G_{best}$  of the whole population in the iteration K were recorded. The positions and velocities of each particle are updated by tracking the two optimal positions.

The velocity and position updating formulas of PSO algorithm were first proposed by Kennedy and Eberhart [1], and improved by Shi and Eberhart et al.<sup>[5-6]</sup>, forming an algorithm model recognized by many scholars. The position and velocity updating formulas of the particle  $i$  in the D dimensional coordinates are shown in equations (1) and (2) :

$$v_{id}^{k+1} = \omega v_{id}^k + c_1 r_1 (P_{id}^{best} - x_{id}^k) + c_2 r_2 (G_d^{best} - x_{id}^k) \quad (1)$$

$$x_{id}^{k+1} = x_{id}^k + v_{id}^{k+1}, \quad d = 1, 2, \dots, D \quad (2)$$

In the equation,  $\omega$  is the inertial weight coefficient;  $c_1$  and  $c_2$  are learning factors greater than

zero, and also known as acceleration constants;  $r_1$  and  $r_2$  are random numbers which are uniformly distributed in the region  $[0,1]$ . Shi and Eberhar suggest that  $\omega$  equals 0.7298, and both  $c_1$  and  $c_2$  equal 1.49618<sup>[5]</sup>. In addition, the movement of particles can be limited by setting the position range

$$X_i \in [X_{min}, X_{max}] \text{ and velocity interval } V_i \in [V_{min}, V_{max}].$$

## 2.2 Advantages and disadvantages of PSO

The PSO algorithm has weakness in local development ability mainly because of the low convergence accuracy, whose reason is that each particle moves through the position and velocity formulas in the algorithm. And these formulas take into account the global search capability and local development capabilities of the particle swarm. Global search capability requires a diversity of particle swarms, while local development capability requires fast clustering particles, which appears to be a contradiction in the same iteration cycle. Most of the improved PSO algorithms find a balance between the two capabilities, resulting in the PSO algorithm being able to adaptively balance the two capabilities in different iteration loops. In the process of searching the global optimization, the first requirement is searching the global optimal region, followed by the global best solution. That is, only if the algorithm has excellent global search capability, it can locate the global optimal region and can accurately search the global best solution. The PSO algorithm has excellent global search capabilities, but its local development capability is slightly inadequate. Therefore, this paper adopts wolf pack search(WPS) to try to relieve the coupling between global search capability and local development capability of PSO algorithm, to ensure the PSO's excellent global search capability, while optimizing the PSO's local development capabilities.

## 3. Particle swarm optimization algorithm based on wolf pack search based

### 3.1 Wolf pack search

Wolf pack search(WPS) is a kind of excellent local development algorithm. Take the search for the maximum value for example, briefly describe the process of WPS. In a D-dimensional hunting space  $\Omega \subseteq \mathbb{R}^D$ , the position of the wolf herd of size N in space is expressed as:  $x_i = \{x_{i1}, x_{i2}, \dots, x_{iD}\}$ , and  $i = \{1, 2, \dots, N\}$ ,  $x_i \in \Omega$ ; The prey concentration that the wolf sensed is  $y_i = f(x_i)$ ; The wolf that finds the greatest prey concentration  $y_{best} = \max\{y_1, y_2, \dots, y_N\}$  is considered prey  $x_{best}$ ; The direction of prey (maximum concentration direction) is  $e = (x_{best} - x_i) / |x_{best} - x_i|$ ; The process by which wolves hunt prey can be abstracted as equations (3) :

$$x_i = x_i + step \times (x_{best} - x_i) / |x_{best} - x_i| \quad (3)$$

In the equation,  $step$  is the moving length of each step of the wolf.

### 3.2 The design of WPS-PSO

The idea of improvement is as follows: In the PSO algorithm, the global optimal particle  $I$  is regarded as prey  $T$  instead of moving according to the formula (1~2), and M artificial wolves  $W = \{w_1, w_2, \dots, w_M\}$  are randomly generated in the surroundings, where the position of the  $m$ -th individual worker wolves is  $w_m = (w_{m1}, w_{m2}, \dots, w_{mD})$ . Then, calculate the objective function value  $Z_m = f(w_m)$  of artificial wolf  $m$ . If the objective function value of the artificial wolf  $m$  is superior to the target function value  $Y_T$  of the prey  $T$ , the roles exchange. Then, the artificial wolf  $W$  sieges the prey  $T$  according to the formula (5). Similarly, if the objective function value  $Z_m$  of an artificial wolf  $m$  is

better than  $Y_T$ 's during the sieging period, the roles exchange. After the siege, the positions and function values of the prey  $T$  are assigned back to the particle  $i$ , and the rest of the particles continue to do the search work according to the formula (1-2). In the iterative process of the PSO algorithm, if the next globally optimal particle is still particle  $i$ , then the artificial wolf is not required to be randomly generated (3), and the siege should be continued on the basis of the siege of the previous generation.

$$w_{md}^k = x_{id}^k \times [\sin(\gamma) + 1] \quad (4)$$

In the equation,  $m=1,2,\dots,M$ ;  $d=1,2,\dots,D$ ;  $\gamma \in [-0.1, 0.1]$  and  $\gamma$  is a uniformly distributed random number.

$$w_{md}^{k+1} = w_{md}^k + \lambda \times (x_{id}^k - w_{md}^k) \quad (5)$$

In the formula,  $x_{id}$  is the position coordinate of the global optimal particle  $i$  in the  $D$ -th dimension, and the random number  $\lambda$  is distributed uniformly between  $\lambda \in (0, 2)$ . It means that the artificial wolf can not only search in front of the global optimal particle  $i$ , but also can search behind it.

### 3.3 The steps of WPS-PSO

Take the process of solving the global maximum for granted, the steps of the WPS-PSO algorithm are shown as follows.

Step 1: Initialize inertia weight coefficient  $\omega$ , learning factor  $c_1, c_2$ , particle group size  $N$ , number of artificial wolves  $M$ , maximum iteration times  $K_{max}$ , particle velocity upper limit  $V_{min}$  and lower limit  $V_{max}$ , particle space location upper limit  $X_{min}$  and lower limit  $X_{max}$ ;

Step 2: Randomly initialize the position  $X$  and velocity  $V$  of  $N$  particles, and calculate the objective function value  $Y$  of each particle;

Step 3: The objective function value of each particle is set to the historical maximum value  $P_{best}$  of each particle, and the largest objective function value of the particle group is set to the global maximum value  $G_{best}$ ;

Step 4: The global maximum particle  $X_{best}$  is set to prey  $T$ , and  $M$  artificial wolves  $W=\{w_1, w_2, \dots, w_M\}$  are created around it according to formula (3);

Step 5: The artificial wolf performs a partial attack strategy on the prey  $T$ . If the objective function value  $Y_{wi}$  of the artificial wolf  $w_i$  is greater than the target function value  $Y_T$  of the prey  $T$  during the attacking period, roles exchange with each other. The rest of the particles adjust the positions and velocities of themselves according to the formula (1-2). After adjustment, if the objective function value  $Y_{k+1}$  of the particle  $i$  is larger than the value  $Y_k$  before, update the historical maximum value  $P_{best}$  of the particle  $i$ .

Step 6: The value of prey  $T$  is assigned back to  $X_{best}$  and the objection function value  $Y_T$  is assigned back to  $G_{best}$ . If one of the maximum value  $P_{best}$  of the remaining particles is greater than the value  $G_{best}$ , then update  $X_{best}$  and  $G_{best}$ .

Step 7: Do the judgment whether the optimization accuracy meets the requirements or the maximum number of iterations  $K_{max}$  is reached. If so, output  $X_{best}$  and  $G_{best}$ . If not and  $X_{best}$  and  $G_{best}$  are updated in Step 6, go to Step 5. If not and  $X_{best}$  and  $G_{best}$  are not updated in Step 6, go to Step 4.

## 4. Simulation

### 4.1 Test function set

In order to verify the effectiveness of the improvement strategy, this section conducts simulation

experiments on WPS-PSO using 9 standard complex test functions, and compares the results with PSO,  $\theta$ -PSO [8], QPSO[9].

(1) Fixed-dimensional test function

$$f_1(x, y) = x^2 + y^2 + 25(\sin^2 x + \sin^2 y)$$

This function is a two-dimensional multi-peak function, and there are several minimum values in the solution space. Get the minimum value 0 at the (0,0) point. In the equation, the  $x, y \in [-2\pi, 2\pi]$ .

$$f_2(x, y) = \frac{-1}{10^4} \left[ \sin x \cdot \sin y \cdot e^{\left( \left| 100 - \frac{\sqrt{x^2 + y^2}}{\pi} \right| \right)} + 1 \right]^{0.1}$$

The function is a two-dimensional multi-peak inseparable function with inverted umbrella shape. There are a large number of local minimums on the umbrella surface, and four global minimum values are symmetrically distributed at the umbrella top; In  $(\pm 1.3491, \pm 1.3491)$  and  $(\pm 1.3491, \mp 1.3491)$ , the function takes the minimum value of -2.0626. In the equation, the  $x, y$  belong to  $[-10, 10]$ .

$$f_3(x, y) = - \left| \sin x \cdot \cos y \cdot e^{\left( \left| 1 - \frac{\sqrt{x^2 + y^2}}{\pi} \right| \right)} \right|$$

The function is a two-dimensional multi-peak inseparable function. There are four global minimum values around the solution space. In  $(\pm 8.05502, \pm 9.66459)$  and  $(\pm 8.05502, \mp 9.66459)$ , the function takes the minimum value of -19.2085. In the equation, the  $x, y$  belong to  $[-10, 10]$ .

$$f_4(x, y) = - \frac{1 + \cos(12\sqrt{x^2 + y^2})}{0.5(x^2 + y^2) + 2}$$

The function is a two-dimensional multi-peak inseparable function with inverted tornado shape. There are numerous local minimums and strong oscillating characteristic. Get the minimum value -1 at the (0,0) point In the equation, the  $x, y$  belong to  $[-5.12, 5.12]$ .

$$f_5(x, y) = \sum_{i=1}^5 i \cdot \cos[(i+1) \cdot x + i] \cdot \sum_{i=1}^5 i \cdot \cos[(i+1) \cdot y + i]$$

The function is a two-dimensional multi-peak inseparable function. There are 18 local minimum values of -186.7309 symmetrically distributed, 9 global maximum values. In the equation, the  $x, y$  belong to  $[-10, 10]$ .

(2) High-dimensional test function

$$f_6(\vec{x}) = \max(|x_i|)$$

This function is a single-peak multidimensional function with the pyramid shape, which is pyramidal with only one global minimum value, and the surrounding is very confusing. In the (0, 0, ..., 0) point, the function gets the minimum value 0. In the equation,  $x_i$  belongs to  $[-32.768, 32.768]$ ,  $i = 1, 2, \dots, n$ .

$$f_7(\vec{x}) = -20e^{\left(-\frac{1}{5}\sqrt{\frac{1}{n}\sum_{i=1}^n x_i^2}\right)} - e^{\left[\frac{1}{n}\sum_{i=1}^n \cos(2\pi x_i)\right]} + 20 + e$$

This function is multi-dimensional multi-peak nonlinear inseparable function. There are many local minimum values. In the  $(0, 0, \dots, 0)$  point, the function gets the minimum value 0. In the equation,  $x_i$  belongs to  $[-32.768, 32.768]$ ,  $i = 1, 2, \dots, n$ .

$$f_8(\vec{x}) = \sum_{i=1}^n \frac{x_i^2}{4000} - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$$

This function is multi-dimensional multi-peak nonlinear inseparable function. It has a bowl-like shape and has a large number of local minimums which uniformly distributed. There are a large number of local minimums around the global minimum, which are easy to make the search fall into a local oscillation. In the  $(0, 0, \dots, 0)$  point, the function gets the minimum value 0. In the equation,  $x_i$  belongs to  $[-600, 600]$ ,  $i = 1, 2, \dots, n$ .

#### 4.2 The experimental setup

According to references [4-5, 8-9], set experimental parameters as follows. The herd number N of PSO,  $\theta$ -PSO and QPSO takes 50. The herd number N of WPS-PSO takes 40 and the number M of artificial wolves takes 10. The learning rate  $c_1$  and  $c_2$  of PSO,  $\theta$ -PSO and WPS-PSO both take 1.7 and the inertia weight coefficient  $\omega$  takes 0.6. The contraction-expansion coefficient of QPSO takes 0.8. For the fixed-dimension test function experiment, the maximum number of iterations  $K_{max}$  takes 100; for the high-dimensional test function, n takes 30, and  $K_{max}$  takes 500.

In the same platform, 50 simulation calculations were performed for each test function, and the results of 50 simulation calculations were statistically analyzed. The performance of the algorithm is compared with the statistical characteristics such as the optimal value, the average value (mainly reflecting the convergence of the algorithm), the standard deviation (mainly reflecting the robustness of the algorithm), and the average consumption time (in seconds).

#### 4.3 Comparisons and analysis of results

The contrast of the optimization results of fixed-dimensional test function sets between WPS-PSO and PSO,  $\theta$ -PSO, and QPSO is shown in Table 1 and figure 1-5 shows the optimization evolution curves of 50 simulations for the four algorithms. According to the chart and figures we can get the following results. For the function  $f_1$ , the average convergence speed and accuracy of the four algorithms are quite similar and the WPS-PSO has slightly superior to other algorithms. For the function  $f_2$ , all four algorithms can get the global minimum value -2.0626. However, the standard deviation accuracy of WPS-PSO is 4 to 7 orders of magnitude higher than the other three algorithms, and its average convergence speed and robustness are better than the other three. For the function  $f_3$ , only the average of WPS-PSO and  $\theta$ -PSO can accurately locate the global minimum value -19.2085. According to Fig.3, both of their average convergence speeds are superior to the speeds of PSO and QPSO. In addition, the performance of WPS-PSO is best. For function  $f_4$ , only the average of WPS-PSO and  $\theta$ -PSO can accurately locate the global minimum value -1. According to Fig.4, both of their average convergence speeds are obviously superior to the speeds of PSO and QPSO. In addition, the standard deviation accuracy of WPS-PSO is 5 orders of magnitude higher than the  $\theta$ -PSO algorithms. For function  $f_5$ , the average of WPS-PSO, PSO and QPSO can accurately locate the global minimum value -186.731. However, the standard deviation accuracy is 5 to 6 orders of magnitude

higher than the PSO and QPSO. According to Fig.5, the average convergence speed of WPS-PSO is better than the other three algorithms.

**Tab.1** Comparison of experimental results on fixed-dimension

Function	Evaluation index	PSO	$\theta$ -PSO	QPSO	WPS-PSO
$f_1$	The optimal value	0	0	1.98E-18	0
	The average value	8.59E-16	8.47E-15	5.37E-15	3.97E-16
	The standard deviation	3.36E-15	3.22E-14	9.20E-15	7.91E-16
	The average time-consuming	0.047	0.044	0.048	0.050
$f_2$	The optimal value	-2.0626	-2.0626	-2.0626	-2.0626
	The average value	-2.0626	-2.0626	-2.0626	-2.0626
	The standard deviation	3.95E-09	4.52E-12	2.93E-11	6.83E-16
	The average time-consuming	0.067	0.063	0.066	0.072
$f_3$	The optimal value	-19.2085	-19.2085	-19.2085	-19.2085
	The average value	-19.1491	-17.5609	-19.2085	-19.2085
	The standard deviation	0.266	2.028	4.44E-14	5.41E-15
	The average time-consuming	0.063	0.064	0.061	0.069
$f_4$	The optimal value	-1	-1	-1	-1
	The average value	-0.990	-1	-0.981	-1
	The standard deviation	0.023	1.52E-06	0.030	1.98E-11
	The average time-consuming	0.061	0.058	0.057	0.065
$f_5$	The optimal value	-186.731	-186.731	-186.731	-186.731
	The average value	-186.731	-140.048	-186.731	-186.731
	The standard deviation	1.86E-05	82.957	6.73E-06	6.49E-11
	The average time-consuming	0.070	0.069	0.065	0.080

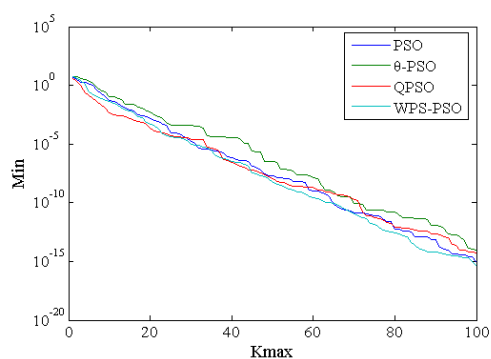


Fig.1 The average convergence curve of  $f_1$

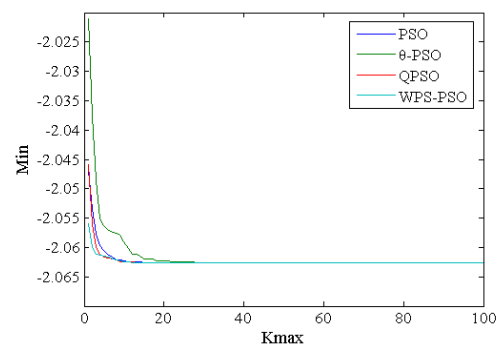
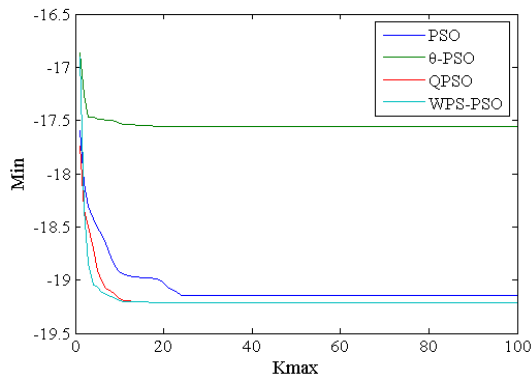
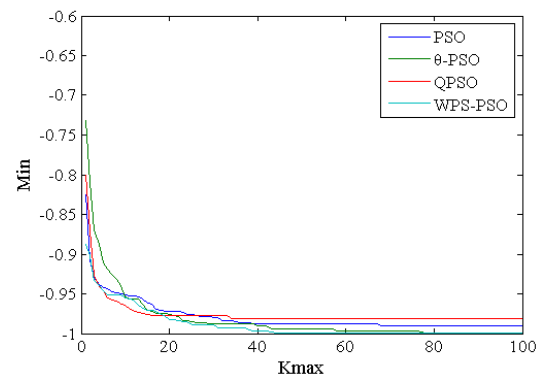
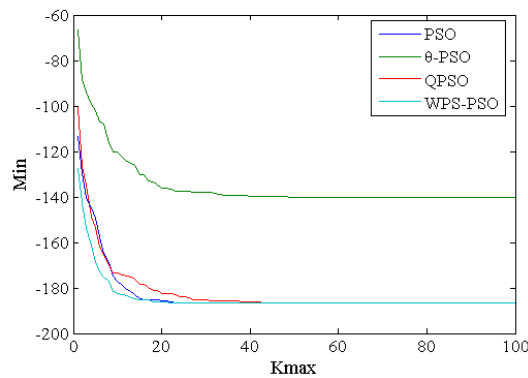
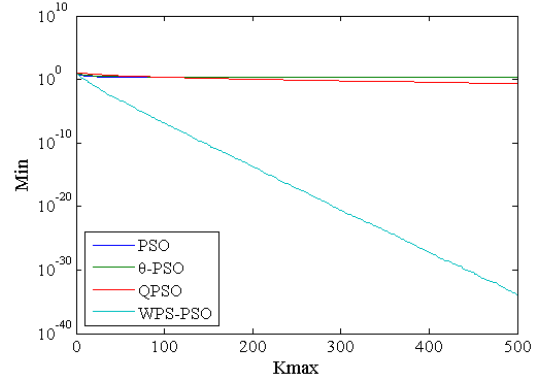


Fig.2 The average convergence curve of  $f_2$

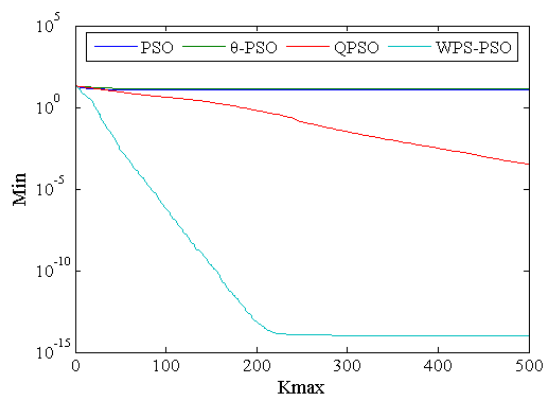
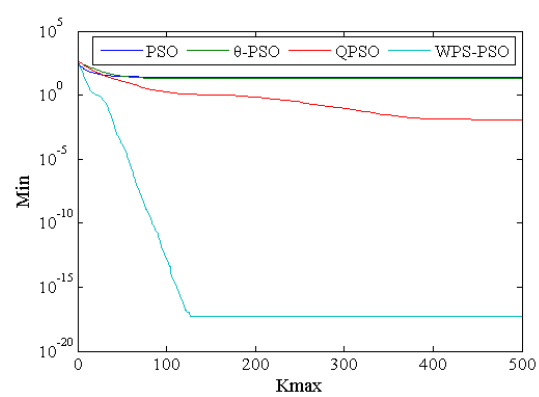


Fig.3 The average convergence curve of  $f_3$ Fig.4 The average convergence curve of  $f_4$ Fig.5 The average convergence curve of  $f_5$ Fig.6 The average convergence curve of  $f_6$ 

The contrast of the optimization results of high-dimensional test function sets between WPS-PSO and PSO,  $\theta$ -PSO, and QPSO is shown in Table 2 and figure 6-8 shows the optimization evolution curves of 50 simulations for the four algorithms. According to the chart and figures we can get the following results. For function  $f_5$  to function  $f_8$ , PSO and  $\theta$ -PSO almost lost the ability of optimization; QPSO has the optimizing ability for function  $f_7$  and function  $f_8$ , but the average convergence speed and accuracy is in sufficient, which is barely able to locate around the global minimum; however, the WPS-PSO algorithm can not only locate the global minimum, but also has faster average convergence speed and higher convergence accuracy. In addition, the comparison of the standard deviation accuracy shows the excellent robustness of WPS-PSO. This indicates that WPS-PSO has advantages for high-dimensional function optimization compared with PSO,  $\theta$ -PSO and QPSO.

**Tab.2** Comparison of experimental results on multi-dimension

Function	Evaluation value	PSO	$\theta$ -PSO	QPSO	WPS-PSO
$f_6$	The optimal value	1.517	2.44E-03	0.104	5.98E-36
	The average value	2.130	2.538	0.193	1.18E-34
	The standard deviation	0.325	2.227	0.063	1.17E-34
	The average time-consuming	0.310	0.325	0.461	0.549
$f_7$	The optimal value	9.036	1.848	7.88E-05	7.99E-15
	The average value	12.012	13.019	3.25E-04	1.01E-14
	The standard deviation	1.426	4.322	2.80E-04	3.34E-15
	The average time-consuming	0.400	0.417	0.543	0.612
$f_8$	The optimal value	7.447	0.717	3.47E-07	0
	The average value	25.991	19.764	1.23E-02	5.55E-18
	The standard deviation	9.055	35.215	1.63E-02	2.48E-17
	The average time-consuming	0.413	0.424	0.548	0.620

Fig.7 The average convergence curve of  $f_7$ Fig.8 The average convergence curve of  $f_8$ 

From the perspective of average time consumption (which can directly reflect the complexity of the algorithm), the average time consumption of WPS-PSO is slightly greater than that of PSO,  $\theta$ -PSO and QPSO, whether it is the optimization of fixed-dimensional functions or high-dimensional functions., but the gap is not particularly obvious. It is calculated that the average time consumption of WPS-PSO is 10.1% more than PSO,  $\theta$ -PSO, and QPSO for the optimization of fixed-dimensional functions. For high-dimensional function optimization, the average time consumption of WPS-PSO is 25.6% more than PSO,  $\theta$ -PSO, and QPSO.

To sum up, the convergence speed and the convergence accuracy are better than PSO,  $\theta$ -PSO and QPSO. Especially for high-dimensional functions, the advantages of WPS-PSO are even more obvious. This indicates that the improved strategy based on wolf pack search can not only maintain the excellent global search ability of the particle swarm optimization algorithm, but also can effectively improve its local development ability and have advantages in solving high-dimensional complex problems.

## 5. Conclusion

In this paper, a new algorithm particle swarm optimization algorithm based on wolf pack search (WPS-PAO) is proposed to improve the weak local development capability of PSO. The simulation results show that not only the strategy has obvious effect on improving the local development capability of PSO algorithm, but also can effectively improve the solution accuracy and convergence speed of PSO for solving fixed-dimensional and high-dimensional complex problems, with excellent performance especially for high-dimensional problems. The improved strategy adds the WPS local search ability to the optimal particles, optimizing the local development ability of the PSO without affecting the PSO's global exploration ability. What's more, the new algorithm provides a new improvement idea for heuristic algorithm.

## References

- [1] Kennedy J, Eberhart R C. Particle swarm optimization[A]. Proc IEEE International Conference on Neural Networks[C]. USA: IEEE Press, 1995.1942-1948.
- [2] Eberhart R C, Kennedy J. A new optimizer using particle swarm theory[A]. Proc Sixth International Symposium on Micro Machine and Human Science [C]. Nago ya, Japan: IEEE Press, 1995.39-43.
- [3] Eberhart R C, Simpson P K, Dobbins R W. Computational Intelligence PC Tools[M]. Boston, MA: Academic Press Professional, 1996.
- [4] Trelea I C. The particle swarm optimization algorithm: convergence analysis and parameter selection[J]. Information Processing Letters, 2003, 85(6):317-325.
- [5] Jiang M, Luo Y P, Yang S Y. Stochastic convergence analysis and parameter selection of the standard particle swarm optimization algorithm[J]. Information Processing Letters, 2007, 102(1):8-16.
- [6] Zhan Z H, Zhang J, Liu O. Orthogonal learning particle swarm optimization[C]. Conference on Genetic and Evolutionary Computation. ACM, 2009:1763-1764.
- [7] Zhang L, Tang Y, Hua C, et al. A new particle swarm optimization algorithm with adaptive inertia weight based on Bayesian techniques[J]. Applied Soft Computing, 2015, 28(C):138-149.
- [8] Zhong W M, Li S J, Feng Q.  $\theta$ -PSO: a new strategy of particle swarm optimization[J]. Journal of Zhejiang University: a volume in English, 2008, 9(6):786-790.
- [9] Xu Y F, Gao J, Chen G C, et al. Quantum Particle Swarm Optimization Algorithm[J]. Applied Mechanics & Materials, 2011, 63-64:106-110.
- [10] Tatsumi K, Ibuki T, Tanino T. Particle swarm optimization with stochastic selection of perturbation-based chaotic updating system[J]. Applied Mathematics & Computation, 2015, 269:904-929.
- [11] Ardizzon G, Cavazzini G, Pavesi G. Adaptive acceleration coefficients for a new search diversification strategy in particle swarm optimization algorithms[J]. Information Sciences, 2015, 299(C):337-378.
- [12] Meng A, Li Z, Yin H, et al. Accelerating particle swarm optimization using crisscross search[J]. Information Sciences An International Journal, 2016, 329(C):52-72.
- [13] Yang C, Tu X, Chen J. Algorithm of Marriage in Honey Bees Optimization Based on the Wolf Pack Search[C]. International Conference on Intelligent Pervasive Computing. IEEE Computer Society, 2007:462-467.