

A Non-need Parameters Optimization Algorithm

Jiakui Zhang*, Lijie Cui, Qing Guo and Haoran Chen

Equipment Management and UAV Engineering College, Air Force Engineering University, Xi'an, Shanxi, 710051, China

*Corresponding author

Abstract—Based on a large number of optimization experiments, the optimal switch probability was discussed. And by the analysis and experiment of local and global pollination, it is found that the global exploration of global pollination was not ideal. In order to integrate the algorithm, this paper designs two kinds of non-need parameters optimization algorithms combined with the idea of orthogonal experiment and based on the following aspects: global exploration, local exploitation, population evolution, and elite reservation. The simulation results of 23 benchmark functions and pressure vessel design problem show that two non-need parameters optimization algorithms have conspicuous advantages in convergence accuracy, convergence speed, stability and engineering optimization.

Keywords—flower pollination algorithm; no parameterization; orthogonal design; function optimization

I. INTRODUCTION

In recent years, many bionic optimization algorithms have been proposed and used in practical engineering widely. In 2012, Yang, a scholar at Cambridge University, proposed a new optimization algorithm named flower pollination algorithm (FPA) inspired by the self-pollination and cross-pollination of flowers in nature[1]. In FPA, cross-pollination on behalf of the global exploration of the algorithm and Self-pollination on behalf of the local exploration of the algorithm are controlled by switch probability. Based on the FPA, Yang proposed a multi-objective flower pollination algorithm (MOFPA)[2][3]. The FPA has fewer initial parameters than classical heuristic algorithms such as Particle Swarm Optimization (PSO) and artificial bee colony algorithm (ABC). However, FPA has some disadvantages such as easy to get into local optimality, poor convergence accuracy. So the global exploration capabilities and local exploration capabilities of FPA still need to be improved. For this reason, some foreign scholars conducted in-depth study of the basic FPA. For example, Abdel-Raouf O et al. proposed an improved pollen algorithm (IFPCH) combined the chaos optimization theory to improve its global exploration ability[4]. For the high-dimensional functions, Wang R et al. introduced the local neighborhood search strategy, the dimension update and evaluation strategy to improve the convergence property of FPA[5]. About the engineering applications, Rodrigues D et al. proposed a Binary Flower Pollination Algorithm (BFPA) to solve the problem of feature selection[6]. El-Henawy I et al. proposed hybrid flower pollination algorithm (FPPSO) to solve the constraint global optimization problem[7]. The above algorithms have improved the optimal capability of FPA. But on the whole, there are still some shortcomings.

In order to further study the optimization characteristics of FPA, this paper, firstly, sets up different switch probability p to test its optimization parameters by four kinds of standard test functions and 23 test functions. Secondly, the global pollination and local pollination were qualitatively analyzed and separated experiment to explore their problems. Then, combined with the orthogonal design and global exploration, local exploration, population evolution and elite reservation, two kinds of Non-need Parameters Optimization Algorithm(NPA) were proposed based on two orthogonal strategies respectively.

II. ANALYSIS OF FPA

A. Flowers Pollination Algorithm

Yang divided pollination of flower into two basic forms: cross-pollination and self-pollination, and abstracted the following four rules[1]:

(1) Biotic and cross-pollination is considered as global pollination process with pollen-carrying pollinators performing Lévy flights.

(2) Abiotic and self-pollination are considered as local pollination.

(3) Flower constancy can be considered as the reproduction probability is proportional to the similarity of two flowers involved.

(4) Local pollination and global pollination is controlled by a switch probability $p \in [0, 1]$. Because of the physical proximity and other factors such as wind, local pollination can have a significant fraction p in the overall pollination activities, and p usually equal to 0.8.

In fact, each flowering plant has more than one flower, and each flower has many pollen gametes in the nature. But Yang assumes that there is only one flower per flowering plant, with only one gamete per flower for algorithm implementation. Each gamete is considered as a candidate solution in the solution space.

$$\left\{ \begin{array}{l} L \sim \frac{\lambda \Gamma(\lambda) \sin(\pi\lambda/2)}{\pi} \times \frac{1}{s^{1+\lambda}} \\ (s \gg s_0 > 0) \end{array} \right. \quad (1)$$

$$x_i^{t+1} = x_i^t + \gamma L(x_i^t - x^{best}) \quad (2)$$

$$x_i^{t+1} = x_i^t + \xi(x_j^t - x_k^t) \quad (3)$$

where x_i^t is candidate solution at iteration t ; L is the strength of the pollination, which essentially is a step size; $\Gamma(\lambda)$ is the standard gamma function, and this distribution is valid for large steps $s > 0$. In all our simulations below, we have used $\lambda = 1.5$. x_{jt} and x_{kt} are pollens from the different flowers of the same plant species.

B. Parameter Analysis

The switch probability p is an important initial parameter of FPA. p is set to 0.8 based on the ratio of cross-pollination and self-pollination of the flowering plants in nature[1]. However, in the actual algorithm optimization, we cannot make sure whether 0.8 is the optimal parameter of p . So, this section provides an in-depth analysis of the parameter sensitivity of FPA.

In order to directly reflect the effect of the switch probability p on the optimization of FPA, according to the different properties of the classical test function, two kinds of single-peak and multi-peak test functions are selected to simulate and optimize based on different switch probability ($p \in [0,1]$), and FPA only performs the strategy of local pollination and greedy evolution. Fig.1-2 shows the Sphere function and Rastrigin function. The pollen population in FPA is 50, and two kinds of classical test functions are optimally calculated respectively by setting $p=(0, 0.1, 0.2, \dots, 1.0)$. Fig.3 shows the optimal

evolution contrast curve, it can clearly indicate the smaller the switch probability p , the better the optimization effect whether for unimodal low-dimensional, unimodal high-dimensional, or multimodal low-dimensional, multimodal high-dimensional function, when $p=0.0$, the effect of FPA achieves the best performance. So, the effect of global pollination based on Lévy flight on the optimization is not very good.

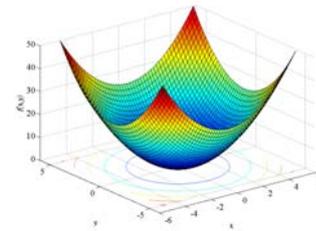


FIGURE I. SPHERE FUNCTION

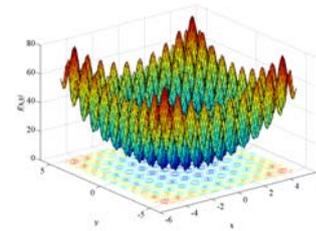


FIGURE II. RASTRIGIN FUNCTION

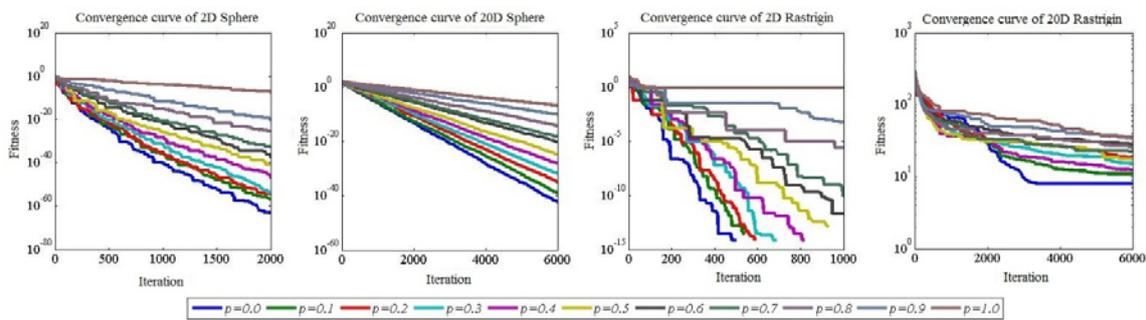


FIGURE III. COMPARISON OF FPA UNDER DIFFERENT SWITCH PROBABILITIES

C. Analysis of Algorithm

For heuristic algorithms, the function of global exploration is to ensure that individuals in the population can appear anywhere in the optimization space with almost equal probability, so that the algorithm can optimize in a larger space to avoid falling into local optimal, so the global exploration is to reflect the randomness of the algorithm. The function of local exploration is to carry on the fine search in the optimal region at present to approach the global optimal point gradually and guarantee the fast convergence of the algorithm. Therefore, local exploration embodies the orientation of the algorithm. For FPA, in pollen algorithm, global pollination is mainly used for global exploration; local pollination is mainly used for local exploration. In global pollination and local pollination, the

location of pollen gametes in the solution space is updated by Eq.(2-3). According to Eq.(2-3), the tendency of pollen gametes position change in two basic pollination processes have visually analyzed. As shown in Fig.4: In each cycle, each pollen gamete that performs global pollination can move only in one direction; The movement direction of pollen gametes that perform local pollination is related to the position of reference pollens, which can be moved in any direction since reference pollens are selected randomly. So, the orientation of global pollination is better than that of local pollination, but the randomness of local pollination is better than that of global pollination. Based on the above analysis, we make the following conjecture: the local pollination of the basic FPA is more suitable for the global exploration.

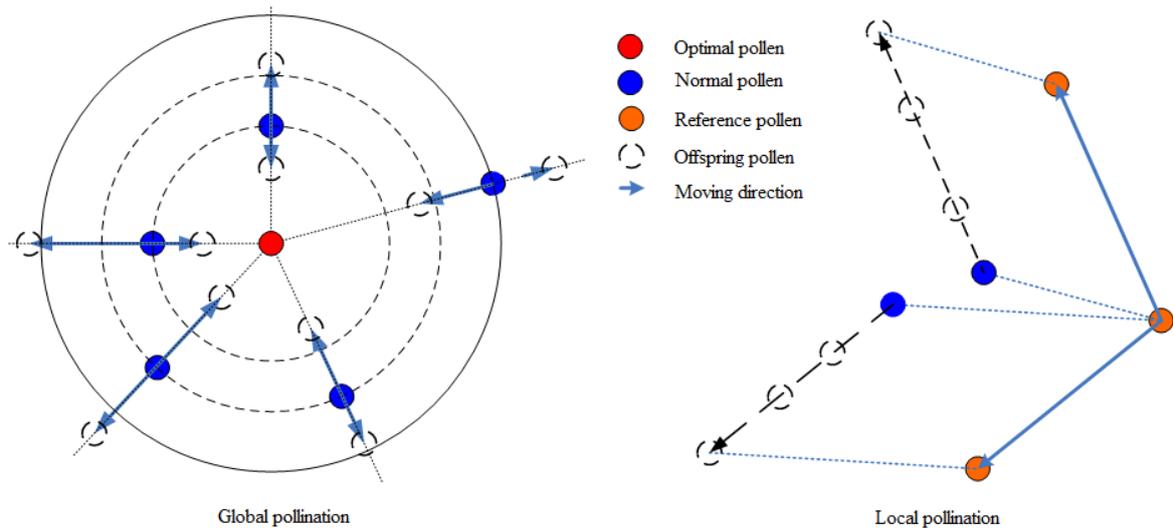


FIGURE IV. CHANGING TENDENCY OF POLLENS IN LOCAL AND GLOBAL POLLINATION

To verify the above conjecture, the greedy evolutionary strategy of the basic FPA is removed, and based on the two-dimensional Sphere function, the global pollination, local pollination and global optimal value updating are separately reserved for optimization computation; for the optimization computation, we have used the pollen gamete population number $N=20$, the maximum number of iterations $K_{max}=100$. Then, the distribution of pollen gametes in the solution space has been observed. As shown in Fig.5, it is the pollen distribution when global pollination is performed separately, and the Fig.6 shows the pollen distribution when local pollination is performed separately. It can be seen that the pollen distribution of local pollination can spread all over the optimal space and the distribution is relatively uniform. The pollen distribution of global pollination is relatively concentrated on the x-axis and y-axis and tends to the center $[0,0]$ of the solution space which is the global minimum of the two-dimensional Sphere function. This more intuitively verify the rationality of the above conjecture: the local pollination of the basic FPA is more suitable for the global exploration.

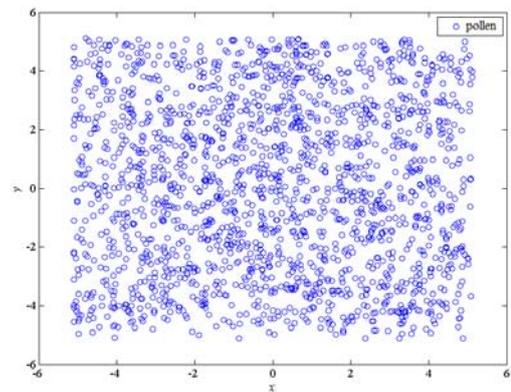


FIGURE VI. POLLENS DISTRIBUTION OF LOCAL POLLINATION

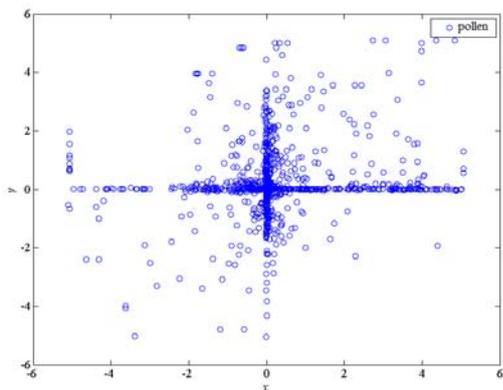


FIGURE V. POLLENS DISTRIBUTION OF GLOBAL POLLINATION

III. P NON-NEED PARAMETERS OPTIMIZATION ALGORITHM

In view of the shortcomings of the basic FPA, the non-need parameters optimization algorithm (NPA) is designed based on the FPA and the idea of orthogonal experimental design.

A. Global Exploration

Because the local pollination in the FPA has excellent randomness and can carry on the optimization in the vast solution space, the pollen gametes of NPA adopt Eq.(4) to perform global exploration.

$$x_i^{t+1} = x_i^t + \xi(x_j^t - x_k^t) \quad (4)$$

$$\xi \in U(0,1)$$

B. Local Exploration

Combined with the Orthogonal Experimental Design (OED), the local exploration of NPA is design by using the

global optimum pollen gametes as the main object to conduct orthogonal design.

1) Orthogonal experimental design

In scientific experiments, there are many interrelated factors affect the test results, so that a large number of comprehensive tests are required to explore the relationship between factors and test results. But when there are too many factors, the number of text will explode. To solve this problem, OED an efficient method to design multi-factor, multi-level research was proposed, which based on the orthogonality of experiment to select “uniform, scattered, neat and comparable” sample points in the comprehensive test.

In scientific experiments, there are many interrelated factors affect the test results, so that a large number of comprehensive tests are required to explore the relationship between factors and test results. But when there are too many factors, the number of text will explode. To solve this problem, OED an efficient method to design multi-factor, multi-level research was proposed, which based on the orthogonality of experiment to select “uniform, scattered, neat and comparable” sample points in the comprehensive test.

```

Input: Q, F
Select a positive integer J by Eq.(6)
Calculate the number of combinations: M=QJ
for k=1 to J do
    j=(Qk-1)/(Q-1)+1
    for i=1 to Qj do
        lij=(i-1)/Qj-1 mod Q
    endfor
endfor
for k=2 to J do
    j=(Qk-1)/(Q-1)+1
    for m=1 to Qj do
        for n=1 to Q-1 do
            l[i+(m-1)(Q-1)+n]=(li*n+lj) mod Q
        endfor
    endfor
endfor
Increment lij by one for 1 ≤ i ≤ M and 1 ≤ j ≤ F
Get the front F columns of l as LM(QF)
Output: LM(QF)
    
```

FIGURE VII. PSEUDO CODE OF CONSTRUCTING ORTHOGONAL TABLE

The test of OED is based on the sample points in the orthogonal table LM(QF), and the best test conditions estimated through factor analysis. Where Q is the number of levels, F is the number of factors and M is the samples size. Pseudo code to generate orthogonal tables is shown in Fig.7[8,10]. Eq.(5) gives the orthogonal matrix of L₄(2³) orthogonal tables.

$$L_4(2^3) = \begin{bmatrix} 1 & 1 & 1 \\ 2 & 2 & 1 \\ 2 & 1 & 2 \\ 1 & 2 & 2 \end{bmatrix} \quad (5)$$

The OED has been successfully applied to GA [9], PSO[10-11] and DE[12], which shows the OED has a good effect on improving the local exploration. Therefore, the improvement of the local exploration in this paper bases on the idea of OED.

2) Local exploration

To improve the local exploration, the orthogonal table LM(QF) is abbreviated as L, where M is the number of offspring generated orthogonally; Q is the number of pollen gametes involved in orthogonal; F is the dimension of pollen gametes which is equivalent to D.

In the new algorithm, the current global optimal pollen gamete (dominated gamete, xp) and another pollen gamete (auxiliary gamete, xs) are orthogonally crossed to generate orthogonal progeny. There are two strategies to select the auxiliary gamete, the one is choosing the pollen closest to the optimal pollen, which bases on the relative distance; the other one is selecting the sub-optimal pollen, which bases on the fitness value. The NPA based on the above two strategies are abbreviated as NPA-d and NPA-f. Fig.8 shows the pseudo code of producing orthogonal generation from the orthogonal table LM(2D), where xL is an M×D matrix, and each row represents an orthogonal progeny.

```

Input: xp, xs, L
for i=1 to M do
    for j=1 to D do
        switch Lij
            case 1: xLj=xpj
            case 2: xLj=xsj
        endswitch
    endfor
endfor
Output: xL
    
```

FIGURE VIII. PSEUDO CODE OF PRODUCING ORTHOGONAL GENERATION

C. Population Evolution Strategy

In heuristic algorithms, the population evolve in the most favorable direction by the guidance of the population evolutionary strategies. The basic FPA uses the greedy evolutionary strategy to evolve the population. But, this evolutionary strategy is found to be too greedy, once the step size is smaller, the pollen gametes can easily fall into the local optimum, which causes the pollen population evolution over.

Therefore, we design a step-by-step evolutionary strategy. In each generation of gametes, the global optimal gamete only executes the local exploration of the OED instead of the global exploration, and the best individual is selected from pollen offspring including the parent to evolve. Then the rest of the pollen gametes execute global exploration. If the offspring is superior to any one of the parents, it will evolve.

D. Elite Reservation Strategy

Elite reservation strategy is designed to reserve the better individuals in the evolutionary process. In the local exploration of the new algorithm, orthogonal strategy is performed by the global optimal gamete, which, therefore, may generate many excellent offspring, some of which will replace the poorly-suited gametes to reserve the elite.

E. New Algorithm Execution Flow

Based on the above ideas for improvement, the implementation steps of the NPA are given

Objective: $Min f(x), x = (x_1, x_2, \dots, x_D)$

Step1: Set the initial parameter: the number of pollen gametes N , the maximum number of iterations K_{max} ;

Step2: Produce the orthogonal table $LM(2D)$ according to Fig.7, initialize the pollen population x_0 and calculate the fitness of each search agent y_0 ;

Step3: Calculate the global minimum and record the optimal solution $xbest$;

Step4: Select the auxiliary gamete xs ;

Step5: The non-global optimum pollen gametes perform global exploration and evolution according to Eq.4;

Step6: The global optimal pollen gamete is orthogonalized and evolved according to Fig.8;

Step7: Replace some poorly-suited pollen gametes with some excellent offspring in Step6;

Step8: Update the global minimum and optimal solution $xbest$;

Step9: If the termination condition is satisfied, output the global optimal solution $xbest$; otherwise, go to Step4.

As can be seen from the algorithm flow, the initial parameters of the new algorithm only involve N and K_{max} , which are indispensable parameters of all heuristic algorithms. Therefore, the new algorithm can be considered as a non-need parameters optimization algorithm, which has some advantages in the optimization calculation.

IV. NUMERICAL SIMULATION AND RESULTS ANALYSIS

In order to verify the validity of the two new algorithm (NPA-d/NPA-f), 23 test functions[13-16] is selected to do the optimization calculation, and a comparison is made with the FPA, the Cuckoo Search Algorithm (CS)[17] and the Firefly Algorithm (FA)[18]. The basic information of fixed dimensional test functions is listed in table 1, which contains F1 to F13. F1 to F5 are single-peak function, with the very flat area around global minimum area, leading to a lot of pollen gametes losing their direction here, therefore these functions can be well used to test the local exploration performance of the algorithm. F6 to F13 are multimodal function, which has a large number of peaks, causing the process of optimization deceptive and making the algorithm stop in local optimal location, so they can effectively test the algorithm of the global exploring ability and the ability to jump out of local optimum. The basic information of multimodal benchmark is listed in table 2, which contains F14 to F23, mainly used to test the local exploring performance and the global exploring performance.

TABLE I. FIXED-DIMENSION BENCHMARK FUNCTIONS

Function		Dim	Range	fmin
Matyas	$f_1(\bar{x}) = 0.26(x^2 + y^2) - 0.48xy$	D=2	[-10,10] ²	Min=0
Branin	$f_2(\bar{x}) = \left(1 - 2y + \frac{\sin 4\pi y}{20} - x\right)^2 + \left(y - \frac{\sin 2\pi x}{2}\right)^2$	D=2	[-10,10] ²	Min=0
Three-Hump Camel	$f_3(\bar{x}) = 2x^2 - 1.05x^4 + x^6/6 + xy + y^2$	D=2	[-5,5] ²	Min=0
Beale	$f_4(\bar{x}) = (1.5 - x + xy)^2 + (2.25 - x + xy^2)^2 + (2.625 - x + xy^3)^2$	D=2	[-4.5,4.5] ²	Min=0
Himmelblau	$f_5(\bar{x}) = (x^2 + y - 11)^2 + (x + y^2 - 7)^2$	D=2	[-6,6] ²	Min=0
Bohachevsky	$f_6(\bar{x}) = x^2 + 2y^2 - 0.3\cos(3\pi x) - 0.4\cos(4\pi y) + 0.7$	D=2	[-100,100] ²	Min=0
Eggcrate	$f_7(\bar{x}) = x^2 + y^2 + 25(\sin^2 x + \sin^2 y)$	D=2	[-2π,2π] ²	Min=0
Levy	$f_8(\bar{x}) = \sin^2(3\pi x) + (x-1)^2 \cdot [1 + \sin^2(3\pi y)] + (y-1)^2 \cdot [1 + \sin^2(2\pi y)]$	D=2	[-10,10] ²	Min=0
Bridge	$f_9(\bar{x}) = -\sin\sqrt{x^2 + y^2} / \sqrt{x^2 + y^2} - \exp[(\cos 2\pi x + \cos 2\pi y)/2] + 0.7129$	D=2	[-10,10] ²	Min=-3.0054
Schaffer	$f_{10}(\bar{x}) = 0.5 + [\sin^2(x^2 - y^2) - 0.5] / [1 + 0.001(x^2 + y^2)^2]$	D=2	[-100,100] ²	Min=0
Box-Betts	$f_{11}(\bar{x}) = \sum_{i=1}^m [e^{-0.1ix} - e^{-0.1iy} - (e^{-0.1i} - e^{-i})z]^2$	D=3	$x, z \in [0.9, 1.2]$ $y \in [9, 11.2]$	Min=0
Colville	$f_{12}(\bar{x}) = 100(x_1 - x_2)^2 + (x_1 - 1)^2 + (x_3 - 1)^2 + 90(x_3 - x_4)^2 + 10.1[(x_2 - 1)^2 + (x_4 - 1)^2] + 19.8(x_2 - 1) \cdot (x_4 - 1)$	D=4	[-10,10] ⁴	Min=0
Hartmann	$f_{13}(\bar{x}) = -\sum_{i=1}^4 c_i \cdot \exp\left[-\sum_{j=1}^6 a_{ij} \cdot (x_j - b_j)^2\right]$	D=6	[0,1] ⁶	Min=-3.3224

TABLE II. MULTIMODAL BENCHMARK FUNCTIONS

Function		Dim	Range	fmin
Sum Squares	$f_{14}(\bar{x}) = \sum_{i=1}^D i \cdot x_i^2$	D=30	[-5.12,5.12] ³⁰	Min=0
Zakharov	$f_{15}(\bar{x}) = \sum_{i=1}^n x_i^2 + \left(\sum_{i=1}^n 0.5i \cdot x_i\right)^2 + \left(\sum_{i=1}^n 0.5i \cdot x_i\right)^4$	D=30	[-5,10] ³⁰	Min=0
Schweffel	$f_{16}(\bar{x}) = \sum_{i=1}^n [(x_0 - x_i)^2 + (1 - x_i)^2]$	D=30	[-10,10] ³⁰	Min=0
Rosenbrock	$f_{17}(\bar{x}) = \sum_{i=1}^{D-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	D=30	[-5,10] ³⁰	Min=0
Multimod	$f_{18}(\bar{x}) = \sum_{i=1}^n x_i \cdot \prod_{i=1}^n x_i $	D=30	[-10,10] ³⁰	Min=0
Styblinski-Tang	$f_{19}(\bar{x}) = \frac{1}{2} \sum_{i=1}^n (x_i^4 - 16x_i^2 + 5x_i)$	D=30	[-5,5] ³⁰	Min=-39.1662D
Ackley	$f_{20}(\bar{x}) = -20 \exp\left(-0.2\sqrt{\sum_{i=1}^D x_i^2/n}\right) - \exp\left[\sum_{i=1}^D \cos(2\pi x_i)/n\right] + 20 + e$	D=30	[-32.768,32.768] ³⁰	Min=0
Griewank	$f_{21}(\bar{x}) = \frac{1}{4000} \sum_{i=1}^D x_i^2 - \prod_{i=1}^D \cos(x_i/\sqrt{i}) + 1$	D=30	[-600,600] ³⁰	Min=0
Rastrigin	$f_{22}(\bar{x}) = 10n + \sum_{i=1}^n [x_i^2 - 10\cos(2\pi x_i)]$	D=30	[-5.12,5.12] ³⁰	Min=0
Penalized	$f_{23}(\bar{x}) = \frac{\pi}{D} \left\{ 10\sin(\pi y_1) + \sum_{i=1}^{D-1} (y_i - 1)^2 \cdot [1 + 10\sin^2(\pi y_{i+1})] + (y_D - 1)^2 \right\} + \sum_{i=1}^D u(x_i, 10, 100, 4)$	D=30	[-50,50] ³⁰	Min=0

$$u(x_i, a, k, m) = \begin{cases} x_i & \text{if } -a < x_i < a \\ -a \cdot k \cdot (-x_i - a)^m & \text{if } x_i < -a \\ a \cdot k \cdot (x_i - a)^m & \text{if } x_i > a \end{cases}$$

In order to make a comparison among these algorithms(only relative comparison) objectively, simulation experiments are conducted on the same MATLAB computing platform, eliminating the difference of running time for the same algorithm on different platforms. According to the literature [1, 17, 18], the parameters of the four optimization algorithms are set, and the result is shown in table 3.

TABLE III. PARAMETERS TABLE OF ALGORITHMS

Parameters	
Common parameters	N=50; For F1-13, $K_{max}=200$, for F14-16, $K_{max}=2000$;
FPA	$p=0.2$; $\lambda=1.5$;
CS	$p_a=0.25$; $\alpha=0.01$; $\beta=1.5$;
FA	$\alpha=0.25$; $\beta=0.2$; $\gamma=1$;

In order to obtain objective simulation calculation results, 50 times optimization, based on statistics, are calculated separately with the selected standard function of FPA, CS, FA and NPA-f. Then, according to the results, we get the statistical indicators such as the optimal value, the mean, the standard deviation and the average time consumption, which are used to measure the performance of the algorithm. The optimal value and average value are mainly used to evaluate the accuracy of the algorithm, the standard deviation is used to evaluate stability of the optimal process, and the average time consumption is used to evaluate the time complexity indirectly.

The comparison of the optimization results of for 23 test functions is listed in table 4. According to the optimization results of fixed dimension test function, we get fourth conclusions. First of all, FPA, CS, FA, NPA-d and NPA-f all can find theoretical solutions of test functions. Secondly, in terms of convergence accuracy, the optimization effect of FPA, CS and FA algorithm is quite similar, but it is obviously inferior to NPA-d and NPA-f. Thirdly, in terms of the optimization stability, the standard deviation of NPA-d and

NPA-f is at least 7 to 21 orders of magnitude smaller than that of FPA, CS and PA in stability. Finally, from the three basic evaluation indexes of the optimal value, mean value and standard deviation of the table, the difference between NPA-d and NPA-f is not significant, which indicates the effectiveness of the two orthogonal strategies. In the high dimensional test function, the whole five algorithms can calculate approximate the theoretical solutions of the single peak test functions, but the convergence of FPA, CS, FA is far less than NPA-d and NPA-f. In addition to F18, the optimization performance of FPA, CS, FA(except F20, F21, F23) is not satisfying. The NPA-d algorithm can find theoretical solutions to all test functions and has high convergence accuracy and stability. However, NPA-f algorithm can find theoretical solutions, equivalent to the effect NPA-d, only for test functions in addition to F20 and F22, proving that the improved algorithm has an advantage the formal algorithm in the performance of high dimensional test function and the strategy based on the adaptive screening of orthogonal auxiliary gametes is less than that based on relative distance.

TABLE IV. COMPARISON RESULTS OF EXPERIMENT FOR TEST FUNCTIONS

F	FPA		CS		FA		NPA-d		NPA-f	
	<i>ave</i>	<i>std</i>								
F1	5.24E-12	5.38E-12	5.26E-07	5.10E-07	6.68E-10	7.65E-10	2.26E-26	4.81E-26	1.53E-25	2.56E-25
F2	6.99E-08	8.55E-08	7.46E-08	7.29E-08	6.34E-06	1.99E-05	2.56E-21	6.74E-21	2.83E-25	5.27E-25
F3	1.03E-10	1.11E-10	9.46E-06	1.07E-05	3.06E-09	3.05E-09	5.60E-26	1.53E-25	1.51E-26	2.11E-26
F4	1.49E-08	2.73E-08	1.04E-04	1.26E-04	2.74E-09	3.28E-09	1.74E-23	2.00E-23	7.03E-25	2.60E-24
F5	1.17E-06	1.16E-06	7.47E-07	1.12E-06	8.50E-08	9.14E-08	1.49E-21	3.94E-21	8.14E-25	2.87E-24
F6	5.52E-07	7.99E-07	2.36E-02	2.64E-02	1.64E-05	1.31E-05	0	0	0	0
F7	8.76E-08	1.29E-07	5.55E-03	5.96E-03	7.39E-08	5.46E-08	1.76E-25	3.14E-25	9.57E-25	1.73E-24
F8	3.96E-08	5.33E-08	5.64E-04	4.93E-04	5.11E-08	4.46E-08	4.09E-25	6.47E-25	4.94E-25	1.46E-24
F9	-2.9619	4.21E-02	-2.8770	0.05	-2.9083	0.2400	-3.0054	0	-3.0054	0
F10	9.94E-06	1.41E-05	2.06E-03	2.44E-03	1.54E-02	1.90E-02	0	0	0	0
F11	8.34E-12	6.95E-12	3.42E-08	2.72E-08	3.83E-11	3.96E-11	2.57E-19	2.74E-19	2.72E-20	5.77E-20
F12	6.65E-01	5.86E-01	3.13	1.47	0.74	1.15	1.62E-10	4.57E-10	1.50E-06	4.50E-06
F13	-3.3205	1.63E-03	-3.28182	1.94E-02	-3.2819	0.10	-3.3224	1.65E-11	-3.3224	1.08E-13
F14	2.05E-02	2.66E-02	3.25E-02	2.09E-02	1.04 E-02	1.39 E-02	5.52E-38	1.71E-37	9.99E-45	3.14E-44
F15	2.86E-01	2.70E-01	0.63	0.37	0.96	0.55	1.04E-19	3.59E-19	7.38E-18	1.25E-17
F16	1.42E-02	1.32E-02	4.02E-02	3.63E-02	3.51E-05	6.68E-06	5.54E-31	6.08E-31	6.45E-31	5.06E-31
F17	25.89	1.02	26.39	1.11	27.54	1.04	6.75E-11	1.65E-10	4.93E-04	9.99E-04
F18	9.88E-25	4.40E-24	2.63E-12	1.20E-11	6.04E-57	2.40E-56	1.42E-184	0	8.69E-294	0
F19	-1066.82	12.9	-1083.15	14.80	-1073.91	28.00	-1174.985	4.67E-13	-1025.136	35.65
F20	2.44	0.6	4.52	0.85	8.39E-03	8.40E-04	2.43E-14	1.75E-14	5.75	2.11
F21	0.41	0.24	0.73	0.13	3.13E-03	1.77E-03	1.11E-16	3.74E-16	6.94E-16	6.37E-16
F22	53.25	7.52	80.76	11.23	30.70	10.06	1.01E-12	3.58E-12	102.78	26.83
F23	2.38E-01	0.22	1.05	0.35	5.67E-06	1.17E-06	1.97E-30	7.52E-30	3.80E-16	1.70E-15

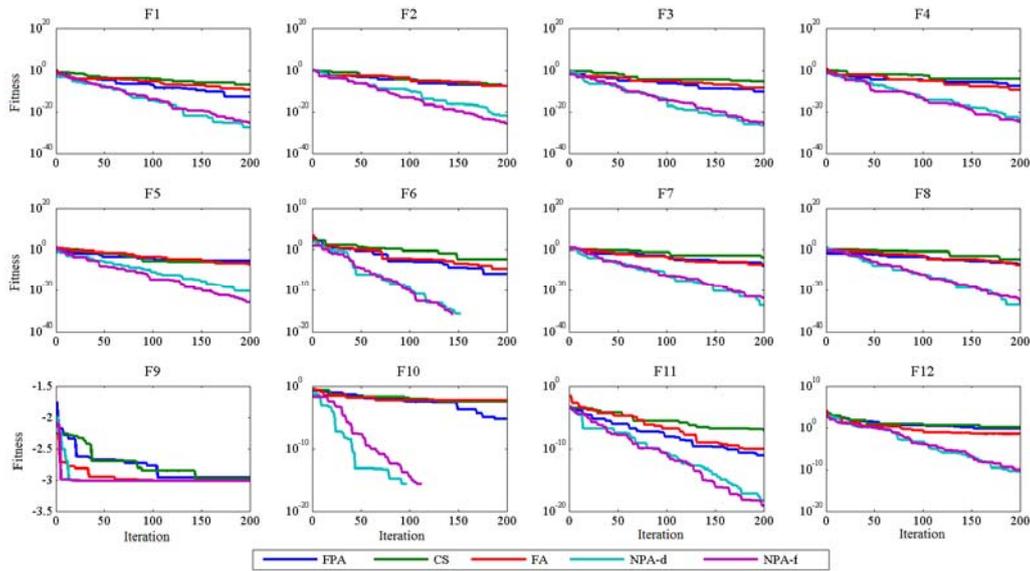


FIGURE IX. EVOLUTION CURVES OF F1-F12

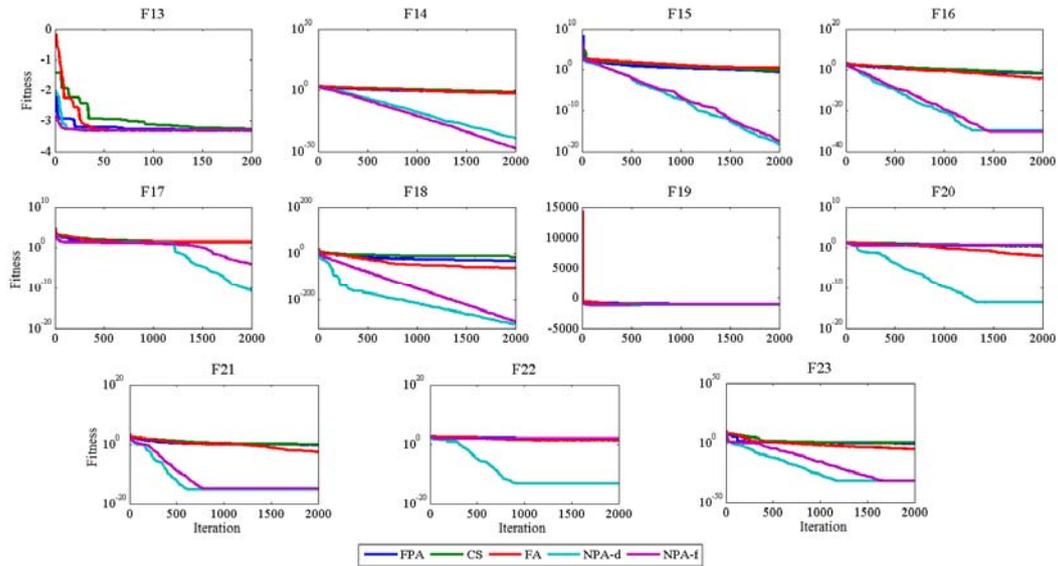


FIGURE X. EVOLUTION CURVES OF F13-F23

In order to find the optimal convergence process of the intuitive reaction algorithm, the optimal evolution contrast curves of 23 test functions are given in Fig.9-10 (Note: the curve in the figure breaks, indicating that the algorithm has converged to zero), from which we can see For NPA-d, the convergence rate of NPA-f is obviously better than that of FPA, CS and FA. For the fixed dimension test functions F1-F13 and the higher-order unimodal functions F14-F16, NPA-d and NPA-f, While the NPA-d algorithm converges faster than NPA-f for the high-dimensional multi-peak function F17-F23.

It is obvious that the optimization convergence rate of NPA-d and NPA-f is significantly better than FPA, CS and FA, from the contrast curves of the 23 test functions in Fig.9-10 (note: the curve fracture in the figure shows that the algorithm has converged to zero). In terms of fixed dimensional functions

F1 to F13 and high dimension single-peak functions F14 to F16, the convergence speed of NPA-d and NPA-f is comparable. However, the optimization convergence rate of NPA-d is a little better than that of NPA-f, in the performance of testing high dimension multimodal function F17 to F23. is given in Tab. 6. from the comparison of the data in Tab.6, which contains the average consumption times of 50 times per test function except F17 and F23, the average time consumption of FA algorithm is the most, followed by CS. FPA is in the middle, NPA-d is less than FPA and NPA-f gets the lowest average consumption time, which indicates that NPA-d and NPA-f is superior to FPA, CS and FA algorithm in time complexity, and is also reflects that NPA-f is better than NPA-d in algorithm complexity.

To sum up, no parameter optimization algorithm not only improves the original global exploring ability of the algorithm and local exploration ability, gets the obvious increase on the convergence precision, convergence speed and calculation stability, but also the time complexity of the algorithm is also got remarkable optimization.

TABLE V. COMPARISON RESULTS OF MEAN TIME OF EXPERIMENT FOR TEST FUNCTIONS

F	Dim	Average time(s)				
		FPA	CS	FA	NPA-d	NPA-f
F1	2	0.2	0.51	1.26	0.18	0.14
F2	2	0.2	0.54	1.25	0.18	0.15
F3	2	0.21	0.55	1.26	0.19	0.16
F4	2	0.2	0.54	1.26	0.18	0.15
F5	2	0.2	0.54	1.25	0.18	0.14
F6	2	0.2	0.54	1.26	0.18	0.14
F7	2	0.21	0.54	1.26	0.18	0.14
F8	2	0.21	0.55	1.26	0.19	0.15
F9	2	0.24	0.59	1.26	0.21	0.18
F10	2	0.2	0.54	1.26	0.18	0.14
F11	3	0.23	0.58	1.29	0.2	0.16
F12	4	0.2	0.55	1.28	0.18	0.14
F13	6	0.45	0.95	1.49	0.43	0.4
F14	30	2.37	6.47	14.10	2.22	1.84
F15	30	2.85	7.16	14.33	2.71	2.55
F16	30	2.5	6.67	14.05	2.35	1.97
F17	30	2.43	6.55	14.22	3.5	3.12
F18	30	2.49	6.61	14.00	2.35	1.98
F19	30	3.17	7.36	14.62	3.11	2.96
F20	30	3.05	7.42	14.46	3.01	2.72
F21	30	3.04	7.41	14.53	3.03	2.68
F22	30	2.71	6.80	14.15	2.51	2.19
F23	30	5.42	11.24	16.40	6.65	6.39

V. PROJECT OPTIMIZATION SIMULATION

The pressure vessel optimal design problem[1] is a constrained optimization problem, which is a classic engineering optimization. The purpose is to design a cylindrical container that meets safety conditions with minimum cost under a given volume and working pressure. The issue involves four important design parameters:

d_1 : Thickness of hemispherical container cover;

d_2 : Thickness of the cylindrical section;

R : Inner radius;

L : Length of the cylinder.

The mathematical description of the expression is as follows:

$$\text{Min } f = 0.6224d_1RL + 1.7781d_2R^2 + 3.1661d_1^2L + 19.84d_1^2R$$

$$s.t. \begin{cases} g_1 = -d_1 + 0.0193R \leq 0 \\ g_2 = -d_2 + 0.0954R \leq 0 \\ g_3 = -\pi R^2L - 4\pi R^3/3 + 1296000 \leq 0 \\ g_4 = L - 240 \leq 0 \end{cases}$$

$$0 \leq d_1, d_2 \leq 99$$

$$10 \leq R, L \leq 200$$

This problem is analyzed by using two non-parametric optimization algorithms, and the results are compared with the calculation results of several classical algorithms in literature [1,19-22], as shown in table 6. It can be seen from the table that NPA-d and NPA-f algorithm have the advantage of solving the optimization design problem of pressure vessels, showing the engineering practicability of the new algorithm.

TABLE VI. COMPARISON RESULTS FOR PRESSURE VESSEL DESIGN PROBLEM

Algorithm	design parameter				Cost
	$d1$	$d2$	R	L	
NPA-d	0.7782	0.3846	40.3196	200	5885.333
NPA-f	0.7782	0.3846	40.3196	200	5885.333
FPA ^[1]	0.8125	0.4375	42.0984	176.6366	6057.714
PSO ^[19]	0.8125	0.4375	42.0913	176.7465	6061.078
GA ^[20]	0.8125	0.4375	42.0974	176.6541	6059.946
DE ^[21]	0.8125	0.4375	42.0984	176.6377	6059.734
ACO ^[22]	0.8125	0.4375	42.1036	176.5727	6059.089

VI. CONCLUSIONS

In order to study the optimization characteristics of the flower pollination algorithm further, the initial parameters of the algorithm are investigated experimentally firstly. It is found that the lower the conversion probability, the better the FPA algorithm is. In order to explain this phenomenon, the global pollination and local pollination process of FPA algorithm are analyzed. It is found that the global exploration ability of the global pollination behavior of pollen gametes is inferior to local pollination and is not suitable for the global exploration of FPA algorithm.

Aiming at the problems above, the idea of orthogonal design is introduced to reintegrate the pollen algorithm in the aspects of global exploration, local exploration, group evolution, elite retention, etc. In addition, two orthogonal strategies are presented and two no parameters optimization algorithms are designed. By comparing the simulation results of 23 standard test functions, it is proved that the two new algorithms have significant advantages in convergence accuracy, convergence speed, stability and time complexity. What's more, compared with the simulation results of pressure vessel optimization design, it is proved that the news algorithms are practical in engineering. However, the new

algorithm also has some stability problems in the optimization of high dimensional complex functions. In the next step, we will solve the issues gradually and extend the algorithm to more engineering applications.

REFERENCES

- [1] YANG X S. "Flower pollination algorithm for global optimization". *International Conference on Unconventional Computation and Natural Computation*, Orléans, France, 2012, pp.240-249.
- [2] YANG X S, KARAMANOGLU M, and HE X. "Multi-objective Flower Algorithm for Optimization". *Procedia Computer Science*, Vol.18,no.1, pp. 861-868. 2013.
- [3] YANG X S, KARAMANOGLU M and HE X. "Flower pollination algorithm: A novel approach for multiobjective optimization". *Engineering Optimization*, Vol.4, no.9, pp. 194-195,2014.
- [4] ABDEL-RAOUF O, ABDEL-BASET M and EL-HENAWY I. "An Improved Flower Pollination Algorithm with Chaos". *International Journal of Education & Management Engineering*, Vol.4, no.2,pp.1-8,2014.
- [5] WANG R and ZHOU Y. "Flower Pollination Algorithm with Dimension by Dimension Improvement". *Mathematical Problems in Engineering*, Vol.2014, no.4, pp.1-9, 2014.
- [6] RODRIGUES D, YANG X S, SOUZA , et al. "Binary Flower Pollination Algorithm and Its Application to Feature Selection". *Studies in Computational Intelligence*, Vol.581, no.1, pp.85-100, 2015.
- [7] EL-HENAWY I, ABDEL-RAOUF O and ABDEL-BASET M. "A New Hybrid Flower Pollination Algorithm for Solving Constrained Global Optimization Problems". *International Journal of Applied Operational Research*, Vol.4, no.2, pp.1-13, 2014.
- [8] AI Zixing, JIANG Zhongyang, WANG Yong and LUO Yidan. "A Novel Constrained Optimization Evolutionary Algorithm Based on Orthogonal Experimental Design". *CHINESE JOURNAL OF COMPUTERS*, Vol.5, pp.855-864, 2010.
- [9] STUBBERUD P A and JACKSON M E. "A Hybrid Orthogonal Genetic Algorithm for Global Numerical Optimization". *International Conference on Systems Engineering*, Las Vegas, USA, 2008, pp.282-287.
- [10] QIN Q, CHENG S, ZHANG Q, et al. "Multiple strategies based orthogonal design particle swarm optimizer for numerical optimization". *Computers & Operations Research*, Vol.60, no.C, pp.91-110,2015.
- [11] ZHANG G and LI Y. "Orthogonal Experimental Design method used in Particle Swarm Optimization for multimodal problems". *Sixth International Conference on Advanced Computational Intelligence*. IEEE, Hangzhou, China, 2013, pp.183-188
- [12] LI Kangshun, ZUO Lei and LI Wei. "Novel differential evolution algorithm based on simplex-orthogonal experimental design". *Journal of Computer Applications*, Vol.1, no.43, pp.143-149, 2016.
- [13] YAO X, LIU Y and LIN G. "Evolutionary programming made faster". *IEEE Transactions on Evolutionary Computation*, Vol.3, no.2, pp.82-102, 2002.
- [14] SALA R, BALDANZINI N and PIERINI M. "Global optimization test problems based on random field composition". *Optimization Letters*, Vol.11, no.5, pp.699-713, 2017.
- [15] MAHMOODABADI M J and NEMATI A R. "A novel adaptive genetic algorithm for global optimization of mathematical test functions and real-world problems". *Engineering Science & Technology An International Journal*, Vol.19, no.4, pp.2002-2021, 2016.
- [16] MIRJALILI S and LEWIS A. "The Whale Optimization Algorithm". *Advances in Engineering Software*, Vol.95, pp.51-67, 2016.
- [17] YANG X S and DEB S. "Cuckoo search: recent advances and applications". *Neural Computing and Applications*, Vol.24, no.1, pp.169-174, 2014.
- [18] YANG X S and HE X. "Firefly Algorithm: Recent Advances and Applications". *International Journal of Swarm Intelligence*, Vol.1, no.1, pp.36-50, 2013.
- [19] HE Q and WANG L. "An effective co-evolutionary particle swarm optimization for constrained engineering design problems". *Engineering Applications of Artificial Intelligence*, Vol.20, no.1, pp.89-99, 2007.
- [20] COELLO C A C and MONTES E M. "Constraint-handling in genetic algorithms through the use of dominance-based tournament selection". *Advanced Engineering Informatics*, Vol.16, no.3, pp.193-203, 2002.
- [21] HUANG F, WANG L and HE Q. "An effective co-evolutionary differential evolution for constrained optimization". *Applied Mathematics & Computation*, Vol.186, no.1, pp.340-356, 2007.
- [22] KAVEH A and TALATAHARI S. "An improved ant colony optimization for constrained engineering design problems". *Engineering Computations*, Vol.27, no.1, pp.155-182, 2010.
- [23] LIANG Y C and JUAREZ J R C. "A novel metaheuristic for continuous optimization problems: Virus optimization algorithm". *Engineering Optimization*, Vol.48, no.1, pp.73-93, 2016.