An Chaotic Firefly Algorithm to Solve Quadratic Assignment Problem

Meng-Wei Guo, Jie-Sheng Wang*, Xue Yang

Abstract—**The firefly algorithm (FA) is a swarm intelligence optimization algorithm based on the glow behavior of fireflies in nature. The classic firefly algorithm has the problem that it is easy to fall into the local optimum and the search accuracy is not high in the later stage. An improved firefly algorithm based on chaos mapping strategy was proposed to solve the quadratic assignment problem (QAP). The egocentricity of chaos just avoids the firefly algorithm from falling into the local optimum, enhances its search ability and solves the convergence prematurely phenomenon of the firefly algorithm. The improved algorithm is compared with genetic algorithm (GA), particle swarm optimization (PSO) algorithm and the classic firefly algorithm to solve the same quadratic assignment problems. The simulation experiments results verify the effectiveness of the proposed algorithm.**

Index Terms—**Firefly algorithm, Quadratic Assignment Problem, Chaos**

I. INTRODUCTION

THE combination optimization refers to find a solution that satisfies the given conditions and makes its objective function value maximum or minimum in a discrete and finite mathematical structure. In general, the combination optimization problems usually have a large number of local extreme points, which are typical NP-complete problems with the characters of non-differentiate, discontinuous, multidimensional, constrained, and highly nonlinear. The quadratic assignment problem (QAP) is widely used in many industrial fields, such as integrated circuit cabling, plant location layout, typewriter keyboard design, job scheduling problem, statistical data analysis, archaeological data sorting, and relay race team sequencing. There are have many research trends and results of quadratic assignment problem in many fields, such as computer aided design, computer graphics, artificial intelligence, image processing, large-scale integrated circuit logic wiring design, computer application science, etc [1-3]. The quadratic assignment problem is a complex discrete combination optimization problem. QAP is not only an NP-hard problem but also does not have a polynomial time approximation algorithm. In general, when the scale of the problem is larger than 20, it is difficult to find its optimal solution by using a classical algorithm in the effective computation time, such as the branch-and-bound method and the cut-plane method. With the development of modern computer technology and the continuous improvement of intelligent optimization algorithms, many heuristic algorithms have been proposed and applied to solve QAP, such as simulated annealing (SA) algorithm, genetic algorithm (GA), ant colony optimization (ACO) algorithm, particle swarm optimization (PSO) algorithm, and Tabu search (TS) algorithm [4-7].

In 2009, the Cambridge scholar Xin-She Yang proposed the firefly algorithm (FA) based on the glow behavior of fireflies in nature [8]. FA mainly adopts the characteristics of firefly luminescence. The firefly individual represents the feasible solution in the solution space, the value of the objective function represents the brightness of the firefly, and the brighter firefly attracts other individuals to move in this direction. The attraction among them is inversely proportional to the distance. If there is not a firefly around for brighter individual, it chooses not to move or randomly change position. After this algorithm was proposed, it attracted a large number of scholars to study it, which has been widely used in continuous optimization, dynamic multidimensional knapsack problems, image compression, multivariable PID controller tuning, community detection in complex networks, network and reliability constrained unit commitment problem, automatic generation control of a combined cycle gas turbine plant and other fields [9-15]. This paper presents a strategy for solving the quadratic assignment problem based on the chaotic firefly algorithm. The effectiveness of the proposed method is verified by simulation experiments.

Manuscript received September 20, 2018; revised December 20, 2018. This work was supported by the the Basic Scientific Research Project of Institution of Higher Learning of Liaoning Province (Grant No. 2017FWDF10), and the Project by Liaoning Provincial Natural Science Foundation of China (Grant No. 20180550700).

Meng-Wei Guo is a postgraduate student in the School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, 114044, PR China (e-mail: 1095536443@qq.com).

Jie-Sheng Wang is with the School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, 114051, PR China; National Financial Security and System Equipment Engineering Research Center, University of Science and Technology Liaoning. (Corresponding author, phone: 86-0412-2538246; fax: 86-0412-2538244; e-mail: wang_jiesheng@126.com).

Xue Yang is a postgraduate student in the School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, 114044, PR China (e-mail: 122270462@qq.com).

II. QUADRATIC ASSIGNMENT PROBLEM (QAP)

A. Mathematical Model of QAP

In 1957, the quadratic assignment problem was proposed by Koopmans and Beckmann in the process of studying the location of economic activities. This problem not only considers the cost of spending at a specific location, but also takes into account the interactions outside the activities, and strives to minimize the cost of applying the device to the same amount of locations. Based on the practical problem and theoretical analysis, a mathematical model for describing quadratic assignment problem (QAP) is derived from different perspectives. There are mainly three mathematical models for QAP.

(1) Koopmans-Beckmann Model

In 1957, when Koopmans and Beckmann were studying the location of economic activities, the problem was not only to consider the cost of spending at a specific location, but also to consider the interactions outside the activities and the cost situation. The proposed model was described as follows:

$$QAP(R,D,C)=\min\left(\sum_{i,j,k=1}^{n} r_{ik}d_{jl}x_{ij}x_{kl}+\sum_{i,j=1}^{n} c_{ij}x_{ij}\right) \quad (1)$$

$$s.t. \sum_{i=1}^{n} x_{ij}=1, \quad i=1,\cdots,n \quad (2)$$

$$\sum_{j=1}^{n} x_{ij}=1, \quad j=1,\cdots,n \quad (3)$$

$$x_{ij}\in\{0,1\}, i=1,\cdots,n; j=1,\cdots,n \quad (4)$$

where, R , D and C are respectively expressed as three matrices, r_{ij} is the unit cost, d_{ij} represents the distance, and c_{ij} represents the extra overhead.

(2) Track Model

When the cost matrix R in Eq. (1) forms a symmetric matrix with the distance matrix D , the trace of the matrix is the sum of the diagonal elements in the symmetric matrix C . So it can be represented as:

$$\sum_{i,j,k=1}^{n} r_{ik}d_{jl}x_{ij}x_{kl}+\sum_{i,j=1}^{n} (RXD^T)_{ij}\, x_{ij}=tr(RXD^T X^T) \quad (5)$$

So the trace model of QAP can be described as:

$$QAP(R,D,C)=\min\left\{tr(RXD^T+C)X^T\right\} \quad (6)$$

(3) Kronecker Inner Product Model

Set $x=vec(X)\in Kn^2$ as a column vector for matrix X , which is described as:

$$vec(RXD)=(D^T\otimes R)vec(x) \quad (7)$$

Eq. (8) can be obtained from Eq. (7).

$$tr(RXD^T X^T)=vec(X)^T vec(RXD^T)=x^T(D\otimes R)x \quad (8)$$

Set $c=vec(C)$ and c is in the same order with x , so the inner product model of QAP and Kronecker can be obtained by:

$$QAP(R,D,C)=\min\left\{x^T(D\otimes R)x+c^T x\right\} \quad (9)$$

The Koopmans-Beckmann model was adopted to solve the quadratic assignment problem with the firefly algorithm. In the described QAP model, $f(1,2,\cdots,n)$ and $l(1,2,\cdots,n)$ represent n factories and n locations respectively. The total cost $F(P)$ is defined in Eq. (10). Each factory should be assigned to a location so as to obtain a distribution scheme P : $f\{1,2,\cdots,n\}\rightarrow l\{1,2,\cdots,n\}$, that is to say the position sequence is $P=\{p_1,p_2,\cdots,p_n\}$ Thus the price $F(P)$ reaches a minimum value.

$$F(P)=\sum_{i=1}^{n}\sum_{j=1}^{n}(r_{ij}d_{p_i p_j}) \quad (10)$$

where, r_{ij} represents the cost among factories, $d_{p_i p_j}$ represents the distance between P_i and P_j , $P\in D^n$ (n is the dimension of searching space), P_i and P_j represent the location number assigned to the factory i and j .

B. Experimental Procedure for Quadratic Allocation Problem

Assume N is the maximum iteration number, the specific experimental procedure is described as follows.

Step 1: Initialize algorithm parameters and set $k=1$.

Step 2: Configure the factory i to the position j according to Eq. (11).

$$P_{ij}=\frac{\eta_{ij}^3}{\sum_{u\in allowed_k}\eta_{iu}^3} \quad (11)$$

Step 3: If $i<n$ (n is the total number of factories), $i=i+1$ and returns to Step (2), otherwise jump to the next step.

Step 4:Calculate the evaluated value of the solutions, update the optimal solution and the average solution. If $k<N$, then $k=k+1$ and return to Step (1), otherwise jump to the next step.

Step 5: Output the optimal solution and the average solution, and end the simulation experiment.

III. Chaotic Firefly Algorithm for Solving Quadratic Assignment Problem

A. Basic Principle of Firefly Algorithm

The firefly algorithm (FA) simulates the feeding and courtship of fireflies based on the bionic principle. The higher the fluorescein, the stronger the attraction. The so-called firefly algorithm randomly places fireflies in the searching space, and each firefly with the same initial value and its own decision domain (viewpoint scope). The movement and the renewal phase of fluorescein of fireflies refer to the iteration of the algorithm.

(1) Renewal Stage of Fluorescein

In the firefly algorithm, its fluorescein is updated according to Eq. (12). The value of the fluorescein is related to its fitness degree. The level of the fluorescein is proportional to the position of the firefly, that is to say it is also directly proportional to its fitness. The fluorescein is updated by the following equation.

$$l_i(t)=(1-\rho)l_i(t-1)+\gamma F(x_i(t)) \quad (12)$$

where, $l_i(t)$ is the fluorescein value of the i th individual in the generation t , $\rho\in(0,1)$ is the fluorescein volatile coefficient, $x_i(t)$ refers to the location of the i th individual after the t th iteration, and $F(x)$ is used to calculate the fitness function.

(2) Movement Strategy of Fireflies

In order to calculate the neighbor set between every two fireflies, if fireflies j and i need to form neighbors, they must satisfy the following conditions: j is within i 's field of view and i 's fluorescein is higher than j . The probability of motion from i to j is calculated according to Eq. (13). The roulette method is adopted. The position is finally updated according to Eq. (14), and the field of view is updated in accordance with Eq. (15).

The mobile probability is calculated by:

$$P_{ij}(t) = \frac{l_j(t) - l_i(t)}{\sum_{k \in N_i(t)} (l_k(t) - l_i(t))} \tag{13}$$

The update equation of location is described as:

$$x_i(t+1) = x_i(t) + S * \left(\frac{x_j(t) - x_i(t)}{\left\| x_j(t) - x_i(t) \right\|} \right) \tag{14}$$

where, $x_i(t) \in R^m$ indicates the position of firefly i in the m th dimensional real-space, $\|\cdot\|$ represents the standard Euclidean distance operator, and $s > 0$ is the move step of firefly. The update of the view field radius of firefly is realized by Eq. (15).

$$r_d^i(t+1) = \min\left\{ r_s, \max\left\{ 0, r_d^i(t) + \beta(n_t - |N_i(t)|) \right\} \right\} \tag{15}$$

where, r is the perceived radius of the firefly, β is the change rate of neighborhood, n is a parameter to control the neighboring fireflies, $|N_i(t)|$ is the number of fireflies in the neighborhood of the individual i .

The firefly algorithm has many parameters, which are described as follows.

(1) ρ is the fluorescence factor of fluorescein. When ρ is equal to 1, it means that the algorithm has no memory, and the value of fluorescein only depends on the fitness of the current position. If the value of ρ is located in [0,1], it indicates the amount of fluorescein in the FA.

(2) γ refers to the increase of the fluorescein value, which indicates the proportion in accordance to the positional fitness.

(3) β is the rate of change of the neighborhood, which must not be too large. If it is too large, it will cause the range to change only between the upper and lower limits.

(4) n_t refers to the neighbor's threshold value. The bigger value will lead to a decrease in the diversity of the swarm. If its value is too small, the individual will be isolated.

(5) l_0 refers to the initial fluorescein value. The firefly will update its own fitness value.

(6) r_s refers to the firefly's perceived radius. Too large r_s will reduce the the searching ability. Too small r_s will make the individual not perceive other fireflies. So r_s is related to the search interval of the objective function.

(7) s is the step size, which is positively related to the accuracy of the solution and the number of iterations.

In 2008, Krishnanand et al. calculated the reference values for various parameters through multiple simulation experiments. These parameters have great significance on the study of firefly algorithms, which are shown in Table 1.

B. Flowchart of Firefly Algorithm

The procedure flowchart of firefly algorithm is shown in Fig. 1. The specific algorithm steps are described as follows.

(1) Initialize the parameters ρ , γ , β , n_t , l_0 and s of FA and the positions of all particles. The dynamic decision range $r_d^i(t) = r_s$ and the firefly position $x_i(t)$ are initialized in the searching space. Calculate the fitness value $F(x_i)$ to obtain the initial optimal position x^* and the optimal value $F(x^*)$, and make the iteration number is 1.

(2) Update the fluorescein value $l_i(t)$ according to Eq. (12). Calculate the neighbor set $N_i(t)$ and the moving probability $P_{ij}(t)$ from i to j according to Eq. (13).

(3) The roulette method is adopted to select the direction of movement. Calculate the position $x_i(t+1)$ of i at $t+1$ iteration number according to Eq. (14) and the decision radius $r_d^i(t+1)$ based on Eq. (15).

TABLE 1. PARAMETERS REFERENCE VALUES OF FIREFLY ALGORITHM

Parameter	ρ	γ	β	n_t	S	l_0
Value	0.4	1	2	5	0.03	5

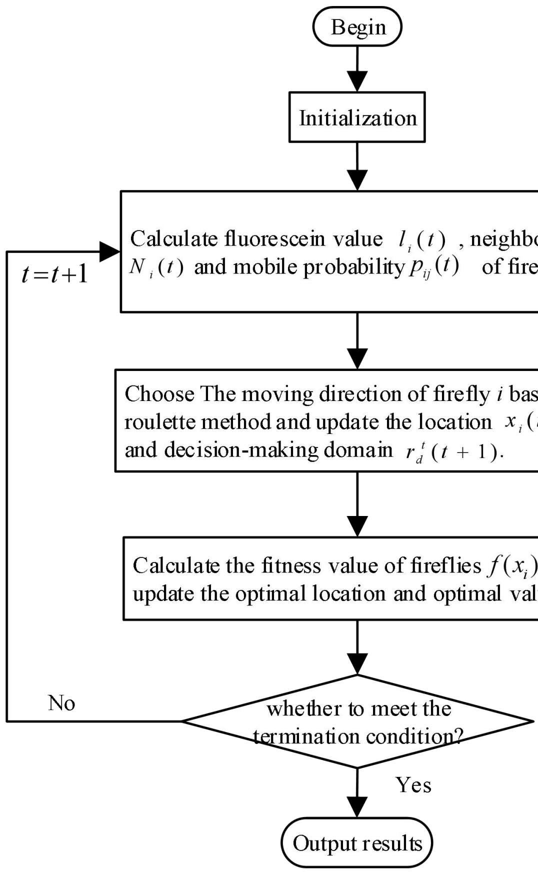


Fig. 1 Flowchart of firefly algorithm.

(4) Calculate the current fitness level $F(x_i)$. If it is better than the set value $F(x^*)$, update x^* and $F(x^*)$.

(5) If the termination condition is satisfied, the iteration is stopped and the output value is the optimal solution x^* with $F(x^*)$. Otherwise, set $t = t + 1$, and go to Step 2 and perform next iteration.

C. Gauss/mouse Chaotic Mapping Strategy

In 1963, the American meteorologist Lorenz give a conclusion that is inconsistent with Laplace's deterministic theory when he simulated the atmospheric turbulence between two infinitely large planes. That is to say, the determined equation can produce random results. The state of chaotic motion is random and is derived from the deterministic equation (internal factor) rather than from randomness (external factors).

Therefore, the ergodicity of chaos enables the variable to traverse each state according to its own situation. The amplification of chaos can cause a small action to play a big role and far exceeds the movement itself alike with the butterfly effect and the turmoil in the stock market. This paper uses Gauss/mouse chaotic mapping strategy and the range of solution set is (0,1). The expression of the chaotic mapping method is described as:

$$j = \begin{cases} 1 & x_i = 0 \\ \dfrac{1}{\mathrm{mod}(x_i, 1)} & otherwise \end{cases} \tag{16}$$

D. Solving Quadratic Distribution Problem Based on Chaotic Firefly Algorithm

For the given n positions and n factories, sort the factories from 1 to n according to the descending order. Then the probabilistic selection based on the roulette method is adopted to allocate the position, that is to say Eq. (17) is used to assign the factory i to position j. After n times, a position sequence with respect to n is obtained, and this solution is the initial solution.

$$P_{ij} = \frac{\eta_{ij}^3}{\sum_{u \in allowed_k} \eta_{iu}^3} \tag{17}$$

where $allowed_k$ represents a collection of factories that have completed allocation, η_{ii} represents the heuristic information of the factory i to factory j, and η_{ii} denotes the heuristic information that has been assigned to the factory.

When the firefly algorithm is applied to the secondary distribution, the position of the firefly is a sequence of n positions, which is a n dimensional vector. Because the solution space is discontinuous, all the distances to the fireflies in the discrete space will also be somewhat different.

Definition 1: Assume two decision-making quantities $X_i = \{x_{i1}, x_{i2}, x_{i3}, \cdots, x_{in}\}$ and $X_j = \{x_{j1}, x_{j2}, x_{j3}, \cdots, x_{jn}\} \in D^n$ represent the positions of two different fireflies i and j. Comparing the components corresponding to the two decision domains, the same is 0 and the difference is 1. The distance between two firefly can be calculated by:

$$\mathrm{distance}(X_i, X_j) = \sum_{k=1}^{n} sign(|x_{ik} - x_{jk}|) \tag{18}$$

$$sign(|x_{ik} - x_{jk}|) = \begin{cases} 0 & x_{ik} = x_{jk} \\ 1 & x_{ik} \neq x_{jk} \end{cases} \quad k = 1, 2, \cdots, n \tag{19}$$

Definition 2: Suppose $A \in D^n$, k is a positive integer, and $N(A,k)$ represents the distance neighborhood from the decision variable A to k, then define:

$$N(A,k) = \left\{ B \,\middle|\, \mathrm{distance}(A,B) < k, B \in D^n \right\} \tag{20}$$

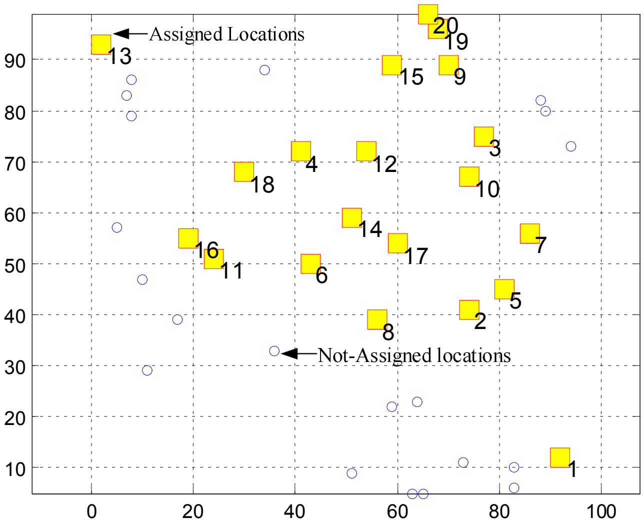
Firefly algorithm has better search capabilities than some other swarm intelligence algorithms, but it is easy to fall into local extreme. The ergodicity of chaos just avoids the algorithm being trapped in local extreme. So the chaotic mapping strategy is adopted to optimize the firefly algorithm's fluorescein and neighborhood radii by combining the global search mechanism of the firefly algorithm so as to obtain the optimal solution.

IV. Algorithm Simulation and Result Analysis

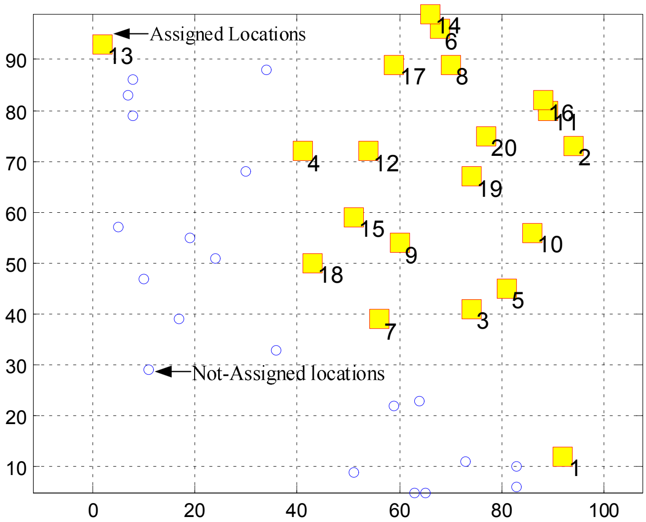
In this paper, genetic algorithm (GA), particle swarm optimization (PSO) algorithm, firefly algorithm (FA) and the proposed chaos firefly algorithm (C-FA) are used to solve the QAP with the same model respectively, so as to examine the ability of each algorithm to obtain the optimal solution and their convergence characteristics. The algorithm parameter settings are shown in Table 2. The optimization results are shown in Fig. 2 and the convergence curves are shown in Fig. 3, respectively. Table 3 shows the optimization performance for each algorithm.

TABLE 2. INITIAL PARAMETERS OF C-FA

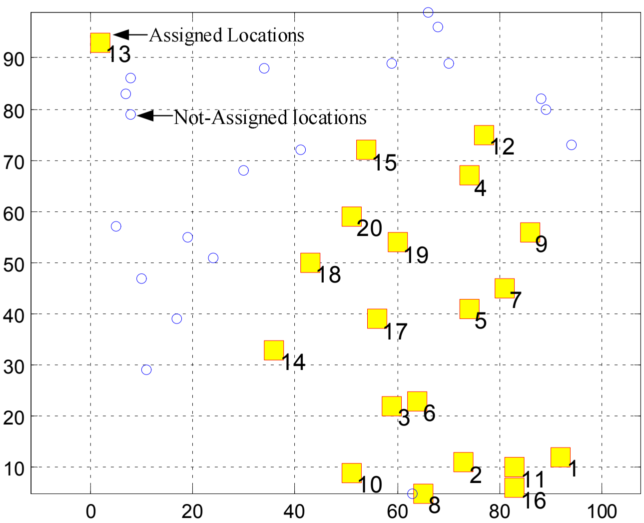
Parameter	Value
Population dimension	20
Fluorescein volatilization factor	0.4
Fluorescein increment	1
Change rate of neighborhood	2
Neighbor threshold	5
Initial fluorescein value	5
Shift step size	0.03
Step size factor	10
Light intensity coefficient	0.2
Maximum iteration number	200



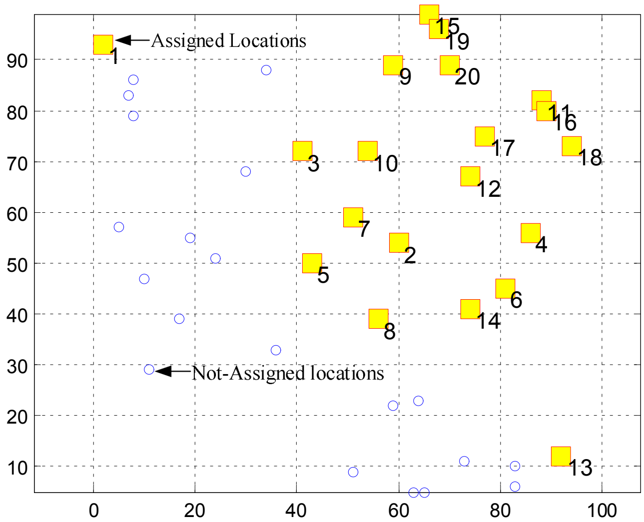
(a) Genetic algorithm



(c) Firefly algorithm



(b) Particle swarm optimization algorithm



(d) Chaotic firefly algorithm

Fig. 2 Simulation results to solve QAP under different algorithms.

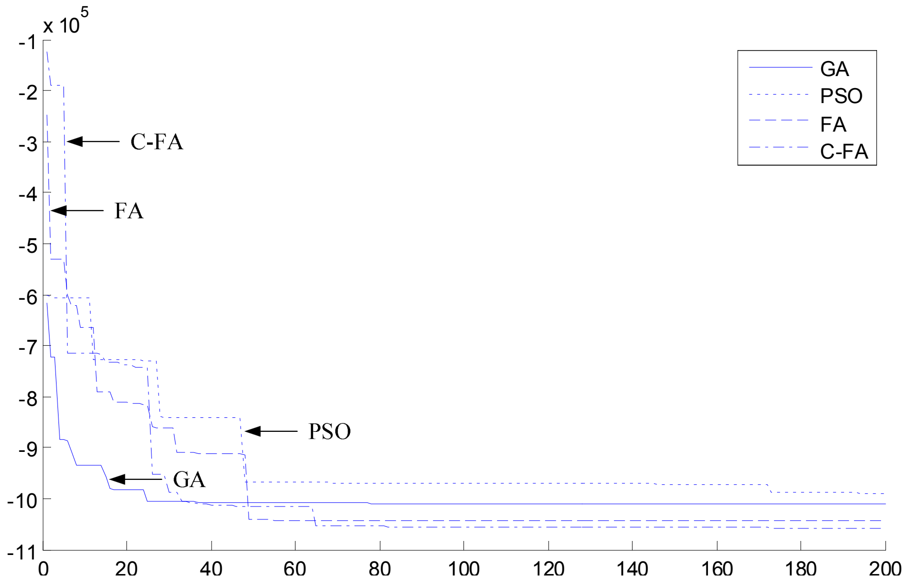


Fig. 3 Simulation comparison curves of four algorithms.

TAB. 3 PERFORMANCE INDICATORS UNDER DIFFERENT OPTIMIZATION ALGORITHM

Algorithm	Optimum	Iteration number to obtain optimum
GA	-1.009e+06	55
PSO	-9.88e+05	173
FA	-1.053e+06	67
C-FA	-1.041e+06	54

The simulation results include the comparison of the convergence effects of the above four different algorithms for solving the quadratic assignment problem and the comparison of the performance indexes of the various algorithms. The simulation results obtained by the C-FA are better than other three algorithms. The following conclusions can be drawn. For solving the secondary distribution problem, the optimization ability of the firefly algorithm is still higher than the rest of the algorithms, but the firefly algorithm itself is easy to fall into the local extreme. The chaotic mechanics is added in the firefly algorithm to form the carrier wave. The characteristics of randomness and ergodicity of chaotic maps can make the firefly algorithm not easily fall into the local extreme and increase the firefly population diversity.

V. CONCLUSIONS

Based on the mathematical model and the characteristics of QAP, the firefly algorithm is applied to solve this problem, which is compared with GA and PSO algorithm to show the superiority of the firefly algorithm. Then the chaotic mapping strategy is introduced into the firefly algorithm and the chaotic firefly algorithm is proposed to solve the convergence premature phenomena of FA. The chaos ergodicity just prevents the algorithm from falling into the local optimum and enhancing its searching ability.

REFERENCES

[1] S. Bougleux, L. Brun, V. Carletti, P. Foggia, B. Gaüzère, and M. Vento, "Graph Edit Distance as a Quadratic Assignment Problem," *Pattern Recognition Letters*, vol. 87, no. C, pp. 38-46, 2016.

[2] J. Sun, Q. Zhang, and X. Yao, "Meta-heuristic Combining Prior Online and Offline Information for the Quadratic Assignment Problem," *IEEE Transactions on Cybernetics,* vol. 44, no. 3, pp. 429-444, 2017.

[3] T. Dokeroglu, and A. Cosar, "A Novel Multistart Hyper-heuristic Algorithm on the Grid for the Quadratic Assignment Problem," *Engineering Applications of Artificial Intelligence,* vol. 52, pp. 10-25, 2016.

[4] E. Lalla-Ruiz, C. Expósito-Izquierdo, B. Melián-Batista, and J. M. Moreno-Vega, "A Hybrid Biased Random Key Genetic Algorithm for the Quadratic Assignment Problem," *Information Processing Letters,* vol. 116, no. 8, pp. 513-520, 2016.

[5] M. Abdel-Baset, L. Abdel-Fatah, H. Wu, and Y. Zhou, "Elite Opposition-flower Pollination Algorithm for Quadratic Assignment Problem," *Journal of Intelligent & Fuzzy Systems,* vol. 33, no. 1, pp. 1-11, 2017.

[6] Y. Ghanou, and G. Bencheikh, "Architecture Optimization and Training for the Multilayer Perceptron using Ant System," *IAENG International Journal of Computer Science*, vol. 43, no.1, pp. 20-26, 2016.

[7] A. P. Song, M. B. Li, X. H. Ding, W. Cao, and K. Pu, "Community Detection Using Discrete Bat Algorithm," *IAENG International Journal of Computer Science*, vol. 43, no.1, pp. 37-43, 2016.

[8] X. S. Yang, "Firefly Algorithms for Multimodal Optimization," *International Symposium on Stochastic Algorithms,* vol. 5792, pp. 169-178, 2009.

[9] X. S. Yang, "Multiobjective Firefly Algorithm for Continuous Optimization," *Engineering with Computers,* vol. 29, no. 2, pp. 175-184, 2013.

[10] A. Baykasoğlu, and F. B. Ozsoydan, "An Improved Firefly Algorithm for Solving Dynamic Multidimensional Knapsack Problems," *Expert Systems with Applications,* vol. 41, no. 8, pp. 3712-3725, 2014.

[11] M. H. Horng，"Vector Quantization Using the Firefly Algorithm for Image Compression," *Expert Systems with Applications,* vol. 39, no. 1, pp. 1078–1091, 2012.

[12] L. D. S. Coelho, and V. C. Mariani, "Firefly Algorithm Approach Based on Chaotic Tinkerbell Map Applied to Multivariable PID Controller Tuning," *Computers & Mathematics with Applications,* vol. 64, no. 8, pp. 2371-2382, 2012.

[13] B. Amiri, L. Hossain, J. W. Crawford, and R. T. Wigand, "Community Detection in Complex Networks: Multi-objective Enhanced Firefly Algorithm," *Knowledge-Based Systems,* vol. 46, no. 1, pp. 1-11, 2013.

[14] K. Chandrasekaran, and S. P. Simon, "Network and Reliability Constrained Unit Commitment Problem Using Binary Real Coded Firefly Algorithm," *International Journal of Electrical Power & Energy Systems,* vol. 43, no. 1, pp. 921-932, 2012.

[15] L. C. Saikia, and S. K. Sahu, "Automatic Generation Control of a Combined Cycle Gas Turbine Plant with Classical Controllers Using Firefly Algorithm," *International Journal of Electrical Power & Energy Systems,* vol. 53, no. 1, pp. 27-33, 2013.

Meng-Wei Guo is received her B. Sc. degree from Liren College of Yanshan University in 2018. She is currently a master student in School of Electronic and Information Engineering, University of Science and Technology Liaoning, China. Her main research interest is mathematical modeling of complex system and intelligent optimization algorithms.

Jie-Sheng Wang received his B. Sc. And M. Sc. degrees in control science from University of Science and Technology Liaoning, China in 1999 and 2002, respectively, and his Ph. D. degree in control science from Dalian University of Technology, China in 2006. He is currently a professor and Master's Supervisor in School of Electronic and Information Engineering, University of Science and Technology Liaoning. His main research interest is modeling of complex industry process, intelligent control and Computer integrated manufacturing.

Xue Yang is a postgraduate student in the School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, 114044, PR China. Her main research interest is mathematical modeling of complex system and intelligent optimization algorithms.