Chaotic Firefly Algorithm with the Optimization Adjustment Strategy for Mobile Robot Path Planning

Dongxue Pang [a], Guolun Guan [b] and Junjie Li [c]

Engineering Research and Development Center for Information Accessibility, Chongqing University of Posts and Telecommunication, Chongqing 400065, China

[a]75335582@qq.com, [b]839731429@qq.com, [c]lijunjie166@126.com

Abstract

Considering the convergence speed and easy trapping into local optimum in robot path planning based on the adaptive firefly algorithm (AFA), a novel algorithm is presented in this paper. Initially, a new chaotic firefly algorithm (CFA) that utilizes chaotic sequences using Lozi's map to tune the control parameters is developed. This scheme avoids the limitation of the adaptive firefly algorithm concerning the local minimum. Subsequently, CFA is enhanced to take advantage of the optimization adjustment strategy (OAS) with the Gauss disturbance to maintain the search capability. Simulation results are compared with those of the adaptive firefly algorithm via a Monte Carlo simulation. It is demonstrated that the proposed CFA-OAS outperforms AFA in terms of the convergence speed and the path length. Moreover, the current study also applies CFA-OAS for robot path planning on robot operating system (ROS). An analysis is conducted to verify feasibility and efficiency of the chaotic firefly algorithm, which is more skillful to pass the narrow area. It can shorten the computation time and search the shorter path.

Keywords

Robot Path Planning; Firefly Algorithm; Lozi's Map; Optimization Adjustment Strategy; Robot Operating System.

1. Introduction

Robot Path Planning is one of the key technologies in robot's offline decision making algorithms. Robot's aim is to find a collision free path from its start position to the target position in this issue [1]. In recent years, with the enlargement of the robot's application range, it has more and more high requirement for path planning technology. Some of new artificial intelligence (AI) technologies gradually are introduced in path planning. Especially, swarm intelligence (SI) algorithms have been widely used in path planning now. For example, the robot path planning can be solved by the ant colony algorithm, Garcia [2] presented a novel method based on ant colony optimization meta-heuristic to solve the robot path planning, and proved it is appropriate for global planners in static and dynamic environment. Some researchers proposed the path planning methods using improved particle swarm optimization methods. Gong [3] put forward a global path planning approach based on multi-objective particle swarm optimization, which is a self-adaptive mutation operation to improve the feasibility of a new path. A phase angle-encoded and quantum-behaved particle swarm optimization (θ-QPSO) was proposed by Fu [4]. It was demonstrated good performance in path planning. Some novel optimization algorithms, such as artificial bee colony (ABC) and artificial fish swarm (AFS), were also used to solve the robot path optimization problem. In [5], a novel artificial bee colony algorithm was improved by a balance-evolution strategy (BES). It is fully utilized to balance between local exploitation and global exploration capabilities. Peng [6] improved the foraging behavior of artificial fish swarm algorithm to enhance the adaptability of the robot global path planning.

Firefly algorithm (FA) is a more promising optimization algorithm put forward by Dr. Yang [7]. It is relatively simple both in theory and in implementation compared with other biological inspired algorithms. The existing research results show that FA had been applied to path planning with good results. Li [8] used firefly algorithm and Bezier curve to locate the shortest feasible (collision-free) path. Some improvements based on firefly algorithm have been put forward in [9,10]. Although these methods were proved effective in solving path planning problems, they inevitably faced some matters in practical applications, such as slow convergence speed, heavy calculation burden, poor stability and easily falling into the local optimum because of a complicated space in the robot working.

A novel path planning method based on the firefly algorithm is presented in this paper. Initially, a new chaotic firefly algorithm (CFA) that utilizes chaotic sequences using Lozi's map to tune the control parameters is developed, which avoids being trapped into the local optimum. Subsequently, the optimization adjustment strategy (OAS) with the Gauss disturbance is for maintaining the search diversity of the newly proposed chaotic firefly algorithm. Superiority of the proposed CFA-OSA in terms of the convergence speed and the path length have been demonstrated utilizing a precise analysis on simulation through a comprehensive comparative study with the adaptive firefly algorithm (AFA) [10]. Moreover, CFA-OSA is compared with AFA in narrow path planning problem. The experimental results show that CFA-OSA outperforms AFA in success rate while passing the narrow area within the shorter time-consuming and path length.

The rest of the paper is organized as follows. The basic firefly algorithm is described in Section 2. Section 3 is devoted to the development of a chaotic firefly algorithm with the optimization adjustment strategy. Section 4 models the robot path planning problem using CFA with OAS. Section 5 provides the results of the numerical simulation and experiment, presenting the comparison of CFA-OSA with AFA. Additionally, this section contains a comprehensive analysis for robot path planning on ROS. Section 6 summarizes the results.

2. Basic firefly algorithm

The firefly algorithm (FA) is a swarm intelligence algorithm, inspired by the flashing behavior of firefly group. Yang formulated FA by three idealized constraints [7] as, 1) All fireflies are no gender-specific, so that one firefly will be attracted to other more larger brightness fireflies. 2) Attractiveness is proportional to their brightness, and for any two fireflies, the less bright one will be attracted to the brighter one; however, the brightness can decrease with distance increases between individuals. If there are no fireflies brighter than a given firefly, it will move randomly. 3) The brightness or attractiveness should be determined by the objective function. For the optimization problem, the luminous intensity is proportional to the value of the objective function.

In FA, the value of objective function is represented by the absolute brightness of fireflies and the solution of the problem to be solved is represented by the position of the fireflies. A firefly's attractiveness is proportional to the light intensity seen by adjacent fireflies. So the attractiveness β of a firefly in terms of Cartesian distance between firefly *i* and firefly *j* as

$$\beta_{ij}(r_{ij}) = \beta_0 e^{-\gamma r_{ij}^2} \tag{1}$$

The movement of a firefly *i* is attracted to another brighter firefly *j* determined by

$$\boldsymbol{x}_i = \boldsymbol{x}_i + \beta_{ij}(r_{ij})(\boldsymbol{x}_i - \boldsymbol{x}_j) + \alpha(rand + 0.5) \tag{2}$$

where the second term is due to the attraction. β_0 represents the biggest attractiveness at $r = 0$. γ is the absorption coefficient, which controls the change in light intensity and determines convergence speed. r_{ij} represents the Cartesian distance between firefly *i* and firefly *j*. The third term is randomization with α being the randomization parameter, which controls the range of movement. The value of rand is a random number generator uniformly distributed in [0, 1]. The pseudo-code for implementing FA can be summarized in Algorithm1.

Algorithm 1: The firefly algorithm
Begin
Objective function $f(x)$, $x = (x_1, ..., x_d)^T$
Initialize population of fireflies x_i (i =1,2,…,n)
Light intensity I_i at x_i is determined by $f(x_i)$
Define light absorption coefficients γ
While (t< MaxIteration)
For i=1:n
For j=1:i
If ($I_j > I_i$)
Move firefly i towards j in d-dimension
End If
Attractiveness varies with distance r via exp[$-\gamma r$]
Evaluate new solutions and update light intensity
End For j
End For i
Rank the fireflies and find the current best
End While
Output result
End Begin

As described above, fireflies are randomly distributed within the search area at the initial period of algorithm running. The distance between each of them is greater, while $\beta_{ij}(r_{ij})$ is small leading to the small range of movement, so the exploring ability of FA is insufficient with slower convergence speed. At the later stage of the algorithm running, fireflies gather around the optimal fireflies, the distance between each other is smaller leading to the bigger $\beta_{ij}(r_{ij})$. At the same time, random motion is still taking place, which is not useful for the convergence of FA [7]. In general, one can improve the search capacity and convergence speed of FA by tuning the parameters γ and α before the run or control them during the run.

3. Chaotic firefly algorithm with the optimization adjustment strategy

The performance of FA is sensitive to the choice of control parameters. It is important to address the value of γ and α in equation (1) and (2). The parameter γ determines the convergence speed of FA, and the parameter α determines the global search capability. In terms of this paper, the absorption coefficient and the randomization parameter in FA are designed by chaotic sequences using Lozi's map. The control parameters are tuned by using the ergodicity property of chaotic sequences, which is beneficial to escape from the local minimum. At the same time, the optimization adjustment strategy is introduced, which maintains the search capability in optimization procedures.

3.1 Chaotic firefly algorithm

Chaos is a kind of a feature of nonlinear dynamic system. It exhibits bounded unstable dynamic behavior, ergodicity and non-period behavior, which depends on initial condition and control parameters. Chaos with the ergodicity property can be used for enriching the searching behavior and avoiding being trapped into the local optimum in optimization problems. In this paper, the improved FA uses chaotic sequences using Lozi's map to tune γ and α given by constant values in the equation (1) and (2), determining the tradeoff between diversity and convergence rate of FA. Lozi's map has the better ergodicity and the mapping points are more uniform distribution then logistic map [11]. The equations above are modified to

$$\boldsymbol{x}_i = \boldsymbol{x}_i + \beta_0 e^{-\gamma(t) r_{ij}^2} (\boldsymbol{x}_i - \boldsymbol{x}_j) + \alpha(t)(rand + 0.5) \qquad (3)$$

with

$$\gamma_1(t) = 1 - a\,|\,\gamma_1(t-1)\,| + \gamma(t-1) \qquad (4)$$

$$\gamma(t) = b\gamma_1(t-1) \qquad (5)$$

and

$$\alpha_1(t) = 1 - a\,|\,\alpha_1(t-1)\,| + \alpha(t-1) \qquad (6)$$

$$\alpha(t) = b\alpha_1(t-1) \qquad (7)$$

where t is the iteration number. a_1, b_1, a_2, b_2 are control parameters, and $a_1 = a_2 = 1.7$, $b_1 = b_2 = 0.5$, these values suggested by [12]. The value of $\gamma(t)$ and $\alpha(t)$ are greatly changed with the variation of a_1, b_1, a_2 and b_2, and they are normalized in the range [0, 1] to each decision variable in N-dimensional space with chaotic dynamics, which determine $\gamma(t)$ and $\alpha(t)$ behave chaotically in unpredictable patterns, other than stabilize at constant sizes. A very small difference in the initial value (t=1) of $\gamma(1)$ and $\alpha(1)$ can cause large differences in its long-time behavior.

In this context, FA based on Lozi's map can be useful in robot path planning for escaping from local optima due to the ergodic and dynamic properties of Lozi's map. This approach is employed to prevent the premature convergence of FA.

3.2 The optimization adjustment strategy

In order to avoid being trapped into the local optimum, an optimization adjustment strategy (OAS) is introduced. For the position of current optimal fireflies, they already tend to be the optimal solution. Thus Gauss disturbance used in the optimal fireflies can maintain the search diversity in CFA. The optimization adjustment equation as

$$\boldsymbol{x}_{best}^{G} = \boldsymbol{x}_{best} + \boldsymbol{x}_{best} \cdot \eta \cdot N(0,1) \qquad (8)$$

where $\boldsymbol{x}_{best}^{G}$ is the position of the current optimal fireflies after Gauss disturbance, $\boldsymbol{x}_{best}$ is the position of the current optimal fireflies, η is the control parameter. The position updation of a new global optimal firefly in the next iteration as

$$\boldsymbol{x}_{best}(t+1) = \begin{cases} \boldsymbol{x}_{best}^{G}(t), & f(\boldsymbol{x}_{best}^{G}(t)) < f(\boldsymbol{x}_{best}(t)) \\ \boldsymbol{x}_{best}(t), & others \end{cases} \qquad (9)$$

In this paper, the current optimal fireflies uses the optimization adjustment strategy to update the position of the next iterative optimal firefly, which is employed to increase the diversity of fireflies' population, and it is advantageous to jump out of local optimum, improving the search speed in CFA at the same time.

4. Robot path planning based on CFA with OAS

4.1 Modeling the path planning problem using CFA

Robot path planning problem is transformed into a numerical optimization problem, which can be solved by CFA-OSA to find a suitable collision-free path between the start position and the end position. The path is planned by encoding of CFA-OSA firstly, establishing the cost function of path planning and then searching for the optimal path using CFA-OSA. In this case, each firefly is defined as a candidate path, and the dimensions of position for them are a set of route points. The path could be completely made by connecting them sequentially. The number of fireflies stands for the candidate path number. In addition, the brightness of a firefly represents the path quality. If the brightest firefly is found, the optimal path will be obtained.

The other important part is establishing the cost function of path planning. In this paper, the cost function is consist of the path length E_L cross the risk value E_D, which is used to evaluate the path quality and can be calculated as follows

$$E = \omega_1 E_L \cdot (1 + \omega_2 E_D) \tag{10}$$

where $\omega_1 + \omega_2 = 1$. We can search the shortest path by calculating min(E).

The path length can be calculated by the following equation

$$E_L = L_{SP_1} + \sum_{i=1}^{n-1} \Delta L_i + L_{P_n E} \tag{11}$$

where (x_i, y_i), i=1…n-1, is the coordinate of the i-th route point between the start position and the end position. L_{SP_1} is the distance from the start position to the first route point. ΔL_i is the distance from the i-th route point to the (i+1)-th route point, and $\Delta L_i = \sqrt{(x_i - x_{i+1})^2 + (y_i - y_{i+1})^2}$. $L_{P_n E}$ is the distance from the last route point to the end position.

The risk value measures the cost of the robot and the obstacle, which is judged by calculating the average cost value of each obstacle in the environment. The risk value can be calculated as follows

$$E_D = 1/m \cdot \sum_{j=1}^{m} (1 - \Delta L_j / R_j) \tag{12}$$

where (x_i, y_i), j=1…m, is the coordinate of the j-th obstacle. ΔL_j is the distance between the robot and the j-th obstacle. R_j is the radius of the j-th obstacle.

4.2 Robot path planning method base on CFA with OAS

The pseudo-code of robot path planning method based on CFA with OAS is shown in Algorithm 2.

Algorithm 2: Path planning method based on CFA with OAS

```
Begin
    Sort the firefly population x_i (i =1,2,…,n) according to the cost function
    Light intensity I_i at x_i is determined by (10),(11),(12)
    Initialize light absorption coefficients γ, randomization parameter α and
    control parameters a1, b1, a2, b2, η
    While (t< MaxIteration)
      For i=1:n
        For j=1:i
            Calculate γ according to (4) and (5)
            Calculate α according to (6) and (7)
            If (I_j > I_i)
                Move firefly i towards j in d-dimension
            End If
            Attractiveness varies with distance r via exp[−γr]
            Reconstruct a path according to x_j
            Evaluate the path
            Update light intensity of x_j
        End For j
      End For i
      Rank the paths and find the current best
      Use the OAS to adjust the position of the current best path
    End While
    Output the optimal path
End Begin
```

5. Experimental results and analysis

5.1 Simulation research

Evaluate the efficiency of path planning method based on chaotic firefly algorithm (CFA) with optimization adjustment strategy (OAS) by simulation, making a comparison with the adaptive firefly algorithm (AFA) in [10] using MATLAB 7.0 on a standard PC with 2.19 GHz processor, 1.96 GB RAM. The calculating way between simple environment and complex environment is different which helps to test the adaptive capacities of two algorithms in different environment.

The parameters of CFA-OAS are shown in Table 1 and the parameters of AFA are the same as [10]. For the comparable methods, the population size is set at 40 and the maximum iteration is 100. Each method is run via a 50-independent-run simulation. Average path length and success rate of searching a shorter collision free path are as evaluation to measure the performance of two path planning methods.

Table 1. Parameters initialization of CFA-OAS

Parameter	Value	Parameter	Value
γ	0.8	η	0.75
α	0.2	β_0	1
a_1	1.7	a_2	1.7
b_1	0.5	b_2	0.5

1) Simple environment

Fig. 1 shows the map including three obstacles as the simple environment to carry out path planning. Where the position of three obstacles are respectively (4, 4), (8, 6), (8, 2) and the radius is respectively 1.5, 1.5, 1.2. The starting position and the end position are from (1, 1) to (11, 7).

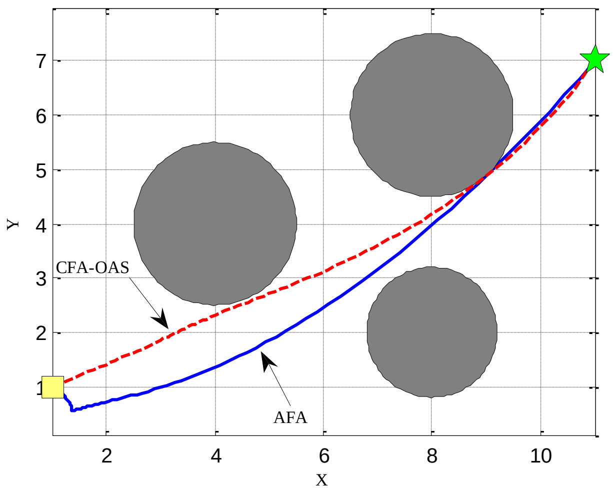

Fig. 1 Path planning in simple environment

Fig. 2 indicates the path length changes with iteration number of two path planning methods in simple environment. From Fig. 2, the method based on CFA-OAS searches the shorter path, and the path length is 11.85 cm. As the same time the CFA-OAS has more quickly convergence speed. It obtains the optimal path after 14 iterations. The reason why this phenomenon appears in CFA-OAS is that chaos with ergodicity property is used to tune control parameters, which enriches the searching behavior and avoids being trapped into local optimum earlier.

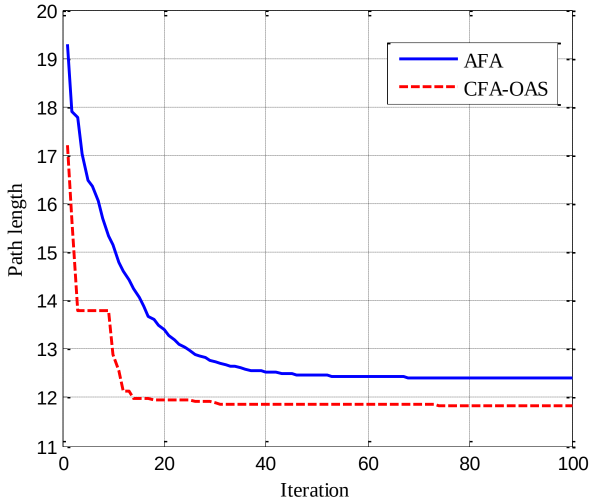

Fig. 2 The path length changes with iteration number in simple environment

The test results are listed in Table 2. The time-consuming of CFA-OAS is similar to AFA. The success rates of two methods in 50 runs are also seen in Tab. 2. The success rate of CFA-OAS is 93%, while the success rate of AFA just achieves 89%. The results suggest that CFA-OAS is more skillful to escape from local trap and search the global optimum. This is because that CFA-OAS adopts the optimization adjustment strategy to maintain the search diversity and enrich the exploitation ability.

Table 2. The test results in simple environment

Algorithm	Iteration number	Time-consuming(s)	Path length(cm)	Success rate(%)
AFA	48	41.05	12.41	89
CFA-OAS	14	43.12	11.85	93

2) Complex environment

Fig. 3 shows the map including eight obstacles as the complex environment to solve path planning. The center coordinates of them are (2.8, 2.8), (4, 5.8), (6.3, 4.5), (8.3, 6.8), (9.5, 2.7), (4.2, -0.3), (6, 1.6) and (8, 0) respectively. The radius of them is respectively 0.8, 0.8, 0.8, 1.1, 1, 0.8, 0.7 and 0.8. The starting position and the end position are also from (1, 1) to (11, 7) as Fig. 3.

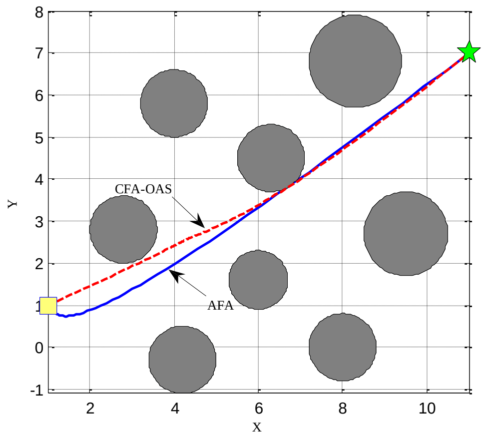

Fig. 3 Path planning in complex environment

The path length changes with iteration number in complex environment and the test results are respectively shown in Fig. 4 and Table 3. It can be found that CFA-OAS still has several advantages on the average path length and the convergence speed than AFA. The success rate in 50 runs of CFA-OAS is much higher than AFA in complex environment, while success rates of both methods are decreased compared with the simple environment.

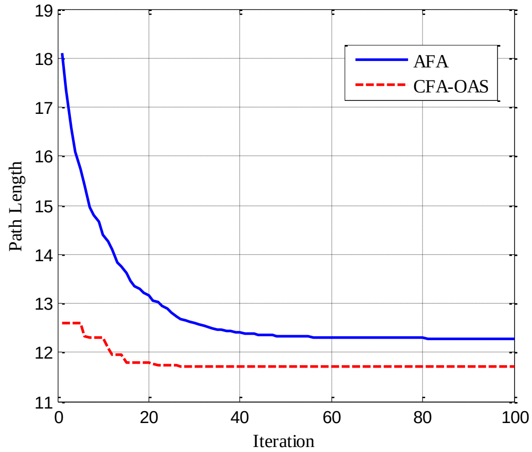

Fig. 4 The path length changes with iteration number in complex environment

Table 3.The test results in complex environment

Algorithm	Iteration number	Time-consuming(s)	Path length(cm)	Success rate(%)
AFA	43	63.17	12.32	81
CFA-OAS	11	65.02	11.71	90

From the two environment tests, we can see that the path planning method based on CFA-OAS has the stronger adaptability with better searching result and quicker convergence speed than AFA in different environment. It is more suitable to apply in robot path planning system.

5.2 Robot path planning using CFA on ROS

In the experiment, AFA algorithm and CFA-OAS algorithm are verified, respectively. Pioneer3-DX robot is equipped with a standard PC with 2.19 GHz processor, 1.96 GB RAM, installing robot operating system (ROS) [13], which collects data using a URG-hokuyo laser ranger. The environment as shown in Fig. 5 is the area of 4.8 m×6 m, consisting of the walls and three obstacles. The map of this environment is build using the slam gmapping node in ROS with resolution 0.050 m/pix as Fig.6, where A and B are the starting position and the target position, respectively. ①,②, ③ are static obstacles, where ①,② are circular obstacles and ③ is rectangular obstacle. The measurement error and the movement error of robot are mixture-Gauss set to be 40cm/3m with probability 0.95. Robot translation speed is set to 0.2m/s. Initial position of robot is unknown, realizing robot self-localization using the amcl node in ROS, and completing robot path planning by two methods on a known environment map finally.

Fig.5 The environment of mobile robot

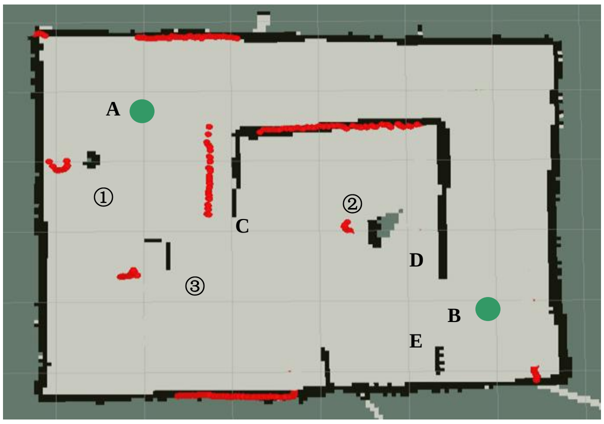

Fig.6 The map of this environment

The paths of the robot from A to B using AFA algorithm and CFA-OAS algorithm are shown as Fig.7 and Fig. 8, respectively. From Fig. 7, AFA algorithm is used, which selects a longer path faced with an obstacle between the two walls; in reverse, the robot searches a shorter path in CFA-OAS and it is more easily to pass a narrow area between the wall C and the obstacle ③. The CFA-OAS determines the distance between the narrow areas more accurately, as shown in Fig. 8, which is that control parameters in CFA-OAS are tuned by chaos with ergodicity property to enrich the searching behavior.

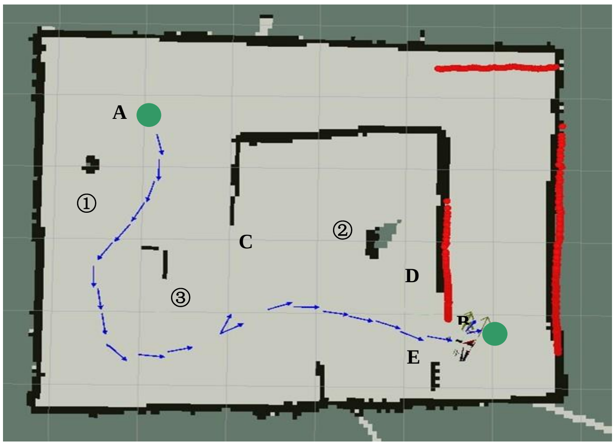

Fig.7 Path of the robot in AFA

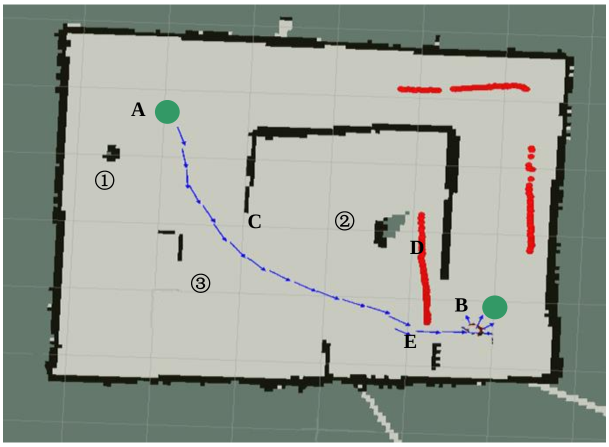

Fig.8 Path of the robot in CFA-OAS

Further on, Table 4 compares the path planning performance of both algorithms. The time-planning in the narrow area of CFA-OAS is superior to AFA. The robot using CFA-OAS can pass the narrow area (between the wall C and the obstacle ③, two walls D and E) more quickly and search the shorter path with 5.23m, leading to shorter time-consuming in the whole path planning from A to B. The success rates of the robot path planning using two algorithms in 50 runs are also shown in Table 4. The success rate of CFA-OAS is 94%, while the success rate of AFA just achieves 82% because of the failure of passing the narrow area.

Table 4. Path planning performance in both algorithms

Algorithm	Time-planning in small area(s)	Time-consuming(s)	Path length(m)	Success rate(%)
AFA	37	123.16	7.81	82
CFA-OAS	15	75.24	5.23	94

6. Conclusion

A novel chaotic firefly algorithm with the optimization adjustment strategy (CFA-OAS) is developed for the robot path planning. The proposed CFA combines the standard firefly algorithm (FA) with the control parameters designed by chaotic sequences using Lozi's map. It plays the important role in firefly algorithm, and the ergodicity property of chaotic sequences is for tuning the control parameters, which is beneficial to jump out of the local minimum and it is more skillful to pass the narrow area. In addition to the optimization adjustment strategy (OAS), the Gauss disturbance is utilized in the optimal fireflies to maintain the diversity of fireflies' population in CFA, avoiding being trapped into local optimum. Finally, simulation and experiment show that the comparison verifies the superiority of the proposed CFA-OAS over the AFA in terms of path length and computation time.

Acknowledgements

This work was supported by the Chongqing Science and Technology Commission (CSTC) (no. cstc2015jcyjBX0066).

References

[1] N. Sadati, J. Taheri. Genetic Algorithm in Robot Path Planning Problem in Crisp and Fuzzified Environments. *IEEE International Conference on Industrial Technology* (Bangkok, Thailand, December 11-14, 2002), Vol. 1 (2002), p.175-180

[2] M. A. P. Garcia, O. Montiel, O. Castillo, et al. Path Planning for Autonomous Mobile Robot Navigation with Ant Colony Optimization and Fuzzy Cost Function Evaluation. Applied Soft Computing, Vol. 9 (2009) No.3, p. 1102-1110.

[3] D. Gong, J. Zhang, Y. Zhang. Multi-objective Particle Swarm Optimization for Robot Path Planning in Environment with Danger Sources. Journal of Computers, Vol. 6 (2011) No.8, p. 1554-1561.

[4] Y. Fu, M. Ding, C. Zhou. Phase Angle-encoded and Quantum-behaved Particle Swarm Optimization Applied to Three-dimensional Route Planning for UA. IEEE Transactions on Systems Man & Cybernetics Part A Systems & Humans, Vol. 42 (2012) No.2, p. 511-526.

[5] B. Li, L. G. Gong, W. L. Yang. An Improved Artificial Bee Colony Algorithm Based on Balance-evolution Strategy for Unmanned Combat Aerial Vehicle Path Planning. Scientific World Journal, Vol. 2014 (2014) No.1, p. 95-104.

[6] J. Peng, X. Li, Z. Q. Qin, et al. Robot Global Path Planning Based on Improved Artificial Fish-swarm Algorithm. Research Journal of Applied Sciences Engineering & Technology, Vol. 5 (2013) No.6.

[7] X. S. Yang. Firefly Algorithms for Multimodal Optimization. Stochastic Algorithms Foundations & Applications, Vol. 5792 (2010), p.169-178.

[8] B. Li, L. Liu, Q. Zhang, et al. Path Planning Based on Firefly Algorithm and Bezier Curve. *IEEE International Conference on Information and Automation* (Hailar, China, July 26-31, 2014), (2014), p.630-633

[9] G. Wang, L. Guo, D. Hong, et al. A Modified Firefly Algorithm for UCAV Path Planning. International Journal of Hybrid Information Technology, Vol. 5 (2012), No. 3.

[10] C. Liu, Z. Gao, W. Zhao. A New Path Planning Method Based on Firefly Algorithm. *Proceedings of the 2012 Fifth International Joint Conference on Computational Sciences and Optimization* (Harbin, China, June 23-26, 2012), (2012), p.775-778

[11] D. H. Liu, S. C. Yuan, Y. Lan, et al. Method of Particle Swarm Optimization Based on the Chaos Map. Journal of XiDian University, Vol. 37 (2010) No.4, p. 764-769. (In Chinese)

[12] L. D. S. Coelho. Reliability-redundancy Optimization by Means of a Chaotic Differential Evolution Approach. Chaos, Solitons & Fractals, Vol. 41 (2009) No.2, p. 594-602.

[13] About ROS. [2015-02-13]. http://www.ros.org/about-ros/.