Robot Path Planning based on Modified Artificial Fish Swarm Algorithm

Yi Zhang [a], Guolun Guan [b] and Dongxue Pang [c]

Research Center of Intelligent System and Robot, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

[a]Gagestriving@126.com, [b]jianyoutianxia@163.com, [c]3180599459@qq.com

Abstract

Path planning is one of the important fields in mobile robot technologies research, it is critical to the efficiency and fidelity of the robot navigation. The solution of path planning is to find a collision-free path from start point to target point in an environment surrounded with obstacle. In this paper, a new modified opposition-based reinforcement learning mechanism is proposed to improve the standard artificial fish swarm algorithm (AFSA), and applied in the path planning problem in real environment. In this modified artificial fish swarm algorithm (MAFSA), an attenuation function is introduced to improve the visual of standard AFSA and get the balance of global search and local search, also an adaptive operator is proposed to enhance the adaptive of step. Besides, we introduce the concept of inertia weight factor in MAFSA inspired by PSO intelligence algorithm to enhance convergence rate and accuracy. Ten well-known benchmark functions are given to illustrate strong searching ability and ideal convergence of MAFSA. The simulation of path planning based on grid-based model is performed to illustrate strong optimization ability in robot navigation of MAFSA, experiment result show the superiority of MAFSA.

Keywords

MAFSA, Path Planning, ROS, Grid-Based Model, Robot Navigation.

1. Introduction

Nowadays, with the popularity and wide application of mobile robot in our daily life, the research on robot path planning attracted more and more attentions of the researchers[1]. Path Planning is an important issue in the field of the robot research, and the development of path planning runs through the whole history of the mobile robot[2]. The goal of path planning is to find an optimal collision-free route from a specified start node to a desired target destination in given environment[3], which cluttered with obstacles. The optimal collision-free path is usually satisfied with some certain optimization criteria such as computation time, distance, energy consumption[4][5], and distance or time is the most commonly adopted criteria[6]. Hence, a lot of algorithms have been applied in robot path planning, such as genetic algorithm, ant colony algorithm, PSO, neural networks, fuzzy logic and so on[7][8]. Artificial fish swarm algorithm (AFSA)[9], one of the state-of-the-art swarm intelligence approaches, which was proposed by Li. Xiaolei in 2002. For simulating the social behavior of fish in nature, it offers new ideals to solve the optimization problems in PID controller parameters optimization, data mining and clustering[10][11]，signal processing[12][13], neutral network classifiers [14][15], and so forth. Therefore, plenty of researchers made some attempts to improve the performance of AFSA. In[16], a new artificial fish swarm optimization is proposed to improve the forging behavior of AFSA, which is closer to reality in order to increase a look at the link ambient. Yao et al. reported a hybrid adaptive artificial fish swarm algorithm (HAAFSA) with the adaptive enhanced prey behavior and the segmented adaptive strategy of artificial fish's view and step, which have been verified on research[17]. The concept of ecological niche [18] is proposed to improve the shortcoming of traditional artificial fish swarm algorithm to obtain optimal solution. In[19], an improved discrete optimization algorithm based on artificial fish swarm (IDAFSA) is presented by improving moving way and swarm and prey behavior to make it discrete, also a strategy is introduced to solve the local optimum. Combined with fractal dimension, the proposed algorithm is applied to attribute reduction problem, the experimental results demonstrate the highly feasibility and effectiveness of the method. In [20], a new path planning method based on an improved AFSA (IAFSA) is proposed and applied in crowd evacuation. By computing the direction and position of artificial fishes, the positon of the crowd in evacuation scenario and the crowd simulated by the artificial fish swarm can be obtained. In addition, to make the simulation of crowd evacuation more realistic, PSO is introduced to calculate the velocity of artificial fishes. Simulation results show that the proposed approach can enhance the fidelity and efficiency of crowd evacuation. To solve the optimization problem of fuzzy neutral network (FNN), a new method based on artificial fish swarm algorithm (AFSA) is proposed in [21]. To the parameter optimization of FNN and the structure optimization of fuzzy rules, AFSA-FNN1 and AFSA-FNN2 is established and realizes the acquisition of parameters of membership function (MF) and the simplification of fuzzy rules. Simulation results of the proposed method applied to path planning shows that the optimized FNN can make the path much smoother.

Despite the fact that all algorithms above are successful in improving the weak point of standard AFSA and gets a positive effect in the Corresponding field in some case, but it still cannot reach the ideal result when it comes to some special optimization problem, such as the robot path planning in complex environment. In this paper, a new modified artificial fish swarm algorithm (MAFSA) is proposed and applied in the path planning problem. In MAFSA, an attenuation function α is introduced to improve the visual of standard AFSA. Also an adaptive operator based on Gaussian distribution function is introduced to enhance the adaptive of step. Besides, inspired by PSO algorithm, we introduce the inertia weight factor β in MAFSA to improve the fish behaviors and balance the exploration and exploitation. The result of test functions shows that MAFSA is more effective, especially when it searches multi-peak and large search space function optimization problem. Simulation of path planning based on grid-based model shows that MAFSA has strong optimization ability in robot navigation.

The remainder of this paper is organized as follows: Section 2 introduces the standard artificial fish swarm algorithms. Section 3 presents several modifications of MAFSA and representation of environment. In Section 4, Optimization ability of MAFSA is evaluated and discussed. Section 5 shows the simulation of MAFSA applied in path planning and discuss the results. Section 6 highlights the superiority of MAFSA and conclusions of this paper.

2. Standard AFSA

Artificial fish (AF) is a fictitious entity of true fish. Generally, fishes go to a position which the food is consisted by fishes' social search behaviors [22]. AF has approximately four social behavior: prey behavior, follow behavior, swarm behavior and leap behavior[23]. We use these behaviors to conduct the analysis and explanation of the problem.

Suppose the state of individual artificial fish is vector X. And we can denote the vector, $X=(x_1,x_2,\ldots,x_n)$ where $x_i\left(i=1,2,\ldots n\right)$ is the optimization variable of fishes. The food concentration in current position of the artificial fish can be expressed as $Y=f\left(X\right)$, where Y is the value of target function. Visual is the range in which AFs can search and step is the maximum length which an AF can move. The distance of two AFs between X_i and X_j can be expressed by Euclidean distance as the follow equation:

$$\mathrm{D}_{i,j}=\left|X_i-X_j\right| \tag{1}$$

The crowd factor $\delta(0<\delta<1)$ is a control parameter to control AFs' crowd around a positon and the best position which AFs ever found will be loaded in bulletin. In the following subsection, behaviors of AFs will be described in detail.

2.1 Prey Behavior

In nature, prey behavior is a basic biological behavior for the fish to find food. We can determine the position X_j in the visual scope of the AF i randomly. X_i is the current position of AF i, and X_j is a random state of its visual distance, and Y is the food concentration which can be expressed as the objective function $Y = f(X)$. The position X_j can be calculated by the follow equation:

$$X_j = X_i + Visual \times rand(0,1) \tag{2}$$

The Y_j and Y_i determine the food concentration of X_i and X_j, if $Y_i < Y_j$, AF moves forward a step from its current position to X_j, which is done by

$$X_i(t+1) = X_i(t) + \frac{X_j(t) - X_i(t)}{\|X_j(t) - X_i(t)\|} \times Step \times rand(0,1) \tag{3}$$

If $Y_i > Y_j$, we select a state X_j random again and judge whether its food consistence satisfy the forward condition. When after try_number times the AF i is not satisfied with the forward condition, concerned AF performs Leap Behavior.

2.1 Swam behavior

In order to keep swarm generality, AFs attempt to move towards the center location in every time of iterations. The central position can be determined as follow equation:

$$X_c = \frac{1}{N}\sum_{1}^{N} X_i \tag{4}$$

Where X_c is the arithmetic average of all AF swarm. And N is the size of the population. Denote n_f as the number of AF swarm in the visual scope of X_c. If $n_f / N < \delta$ and $Y_c > Y_i$, which means the center position of the swarm has a better food consistence and population is not crowd, AF i moves forward a step to the companion center by

$$\mathrm{X}_i(t+1) = \mathrm{X}_i(t) + \frac{X_c - \mathrm{X}_i(t)}{\|X_c - \mathrm{X}_i(t)\|} \times \mathrm{Step} \times \mathrm{rand}(0,1) \tag{5}$$

Otherwise, AF performs prey behavior.

2.2 Follow Behavior

During the moving process of the AF, when a single fish or several ones find food, the neighbor fishes will follow and reach the position quickly. Suppose the current position of the AF i is X_i, and position X_j is the neighbor in its visual scope. Denote n_f as the number of AF swarm in the visual scope of X_c, if $\mathrm{Y}_i < \mathrm{Y}_j$ and $n_f / n < \delta$, AF i moves forward a step to the neighbor X_j. The expression is determined as followed:

$$X_i(t+1) = X_i(t) + \frac{X_j(t) - X_i(t)}{\|X_j(t) - X_i(t)\|} \times Step \times rand(0,1) \quad (6$$

If there are no neighbor around X_i or all of them are dissatisfied the condition, the AF i executes the prey behavior.

2.3 leap Behavior

Leap behavior is the basic behavior to seek food or companions in large ranges, can effectively to prevent local optimization. The AF i performs the leap behavior and changes the parameter to leap out of current position. It chooses a state in the visual and moves toward this state to avoid the local extreme values:

$$X_i(t+1) = X_i(t) + Visual \times \mathrm{rand}(0,1) \tag{7}$$

3. Modified AFSA and environment presentation

In standard AFSA, the visual and step is fixed. The visual determines the search scope, while the step determines the convergence rate and precision of standard AFSA. At the initial state of execution process of standard AFSA, the large visual and step could lead the AF quickly move closer to the global optimization solution, increases the convergence speed. However, after the AF moves close to the target position, the large values of visual and step will make the AF miss the position with better food concentration. If the visual and step are too small, the local search ability increase, while the global search ability decrease. As a result the accuracy of optimal result is reduced. The AF will miss the global optimization solution and fall into local optimization. In order to overcome the shortcoming of standard AFSA, we proposed a new algorithm with some improved parameters and behaviors.

3.1 The new AF population

Opposition-based reinforcement learning is a machine intelligence scheme for learning in dynamic, probabilistic environments. Considering opposite actions simultaneously enables individual states to be updated more than one time shortening exploration and expediting convergence, it has been widely used in machine learning. In this paper, we introduce opposition-based reinforcement learning to optimize the location distribution of artificial fish swarm in the problem space, and to obtain better fitness value compared with the initial one, which can speed up the convergence of AFSA.

Considering a and b as the upper and lower limits of problem space, Suppose the current initial position of the AF i is $X_i(X_i \in [a,b], i \in [1,n])$. Here, a transitional artificial fish swarm population is introduced, which is calculated by the following equation.

$$\begin{cases} x_1 = X_1 \\ x_2 = X_2 \\ x_j = X_i, \quad j \in [1,n] \\ \quad \cdots \\ x_n = X_n \end{cases} \tag{8}$$

The opposite position of x_i is defined as $\tilde{x}_i$, and $\tilde{x}_i$ can be calculated by:

$$\tilde{x}_i = a + b - x_i, \quad i \in [1,n] \tag{9}$$

We can get the opposite position of each artificial fish and construct the opposite artificial fish swarm by Eq. (8). And the food concentration of x_i and $\tilde{x}_i$ are $Y(x_i)$ and $Y(\tilde{x}_i)$. When the opposite population is produced, the new large population, which is consisted with opposite artificial fish swarm population and transitional artificial fish swarm population is produced. In the new modified strategy, the individual of the new large population can be ordered according to their fitness value, the order of the new large population is given in the following Table 1, Where $i \in [1,n]$.

Table 1. The order of the new big population

1	2	3	…	n	n+1	…	2n-2	2n-1	2n
$Y(x_2)$	$Y(\tilde{x}_n)$	$Y(\tilde{x}_i)$	…	$Y(x_4)$	$Y(x_n)$	…	$Y(\tilde{x}_2)$	$Y(x_7)$	$Y(x_i)$
x_2	$\tilde{x}_n$	$\tilde{x}_i$	…	x_4	x_n	…	x_2	x_7	$\boldsymbol{x_i}$

As shown in Table 1, there are 2n individuals in the new large population, the artificial fish individual which has better fitness is ordered at the top, and the individual with worst fitness is ordered at the bottom. Compared with the initial position of the AFs, the accuracy of optimal result which is based on the new large population is higher, but the computation is more intensive, the complexity is also increased. In order to balance the accuracy and complexity, we choose the top n individuals as the new initial population by following equation:

$$\begin{cases} X_1 = x_2 \\ X_2 = \tilde{x}_n \\ X_3 = \tilde{x}_i , \quad j \in [1, n] \\ \quad \cdots \\ X_j = x_4 \end{cases} \tag{10}$$

After the above process completed, the new artificial fish swarm, which is constituted with new individuals, is generated. And the best position of the new artificial fish swarm in each iteration is given as followed:

$$X_{best} = X_1 \tag{11}$$

3.2 Parameter variation on MAFSA

3.2.1 Visual

In order to satisfy the requirement of convergence rate and the balance of global search and local search, we introduce a parameter α to improve the visual of AF. The expression of α is determined as followed:

$$\alpha = exp\left(-25 \times \left(t / Tmax\right)^3\right) \tag{12}$$

Where t denote the current iteration number and T_{max} denote the maximum iteration number. α is the attenuation function, which can minimize the value of α when AF moves close to the target position during the iteration process. The expression of visual is determined as followed:

$$Visual = Visual \times \alpha + Visual_{\min} \tag{13}$$

Where $Visual_{\min}$ is minimum determined by the requirement of the problem space. In the initial stage of the iteration process, the visual is relatively large, and the global search ability is great, the AF can find a better position easily in relatively area. While the visual is getting smaller and smaller, the global search ability decrease and local search ability enhanced.

3.2.2 Step

To balance the accuracy of solution and iteration speed, we introduced Gaussian distribution function to improve the step of AF. The expression of Gaussian distribution function is determined as followed:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{14}$$

Where μ denote the mathematical expectation and σ denote the standard deviation. Here, we set μ to 0. $f(x)$ is an attenuation function, which can minimize the value of $f(x)$ when AF moves close to the target area during the iteration process. We induce Gaussian distribution function as the adaptive operator to enhance the adaptive of Step. The expression of adaptive step is determined as followed:

$$Step = Step \times f\left(t / T_{\max}\right) + Step_{\min} \tag{15}$$

Where $Step_{min}$ is minimum determined by the requirement of the problem space. In the initial stage of the iteration process, the step is relatively large, and the iteration speed is speed up; the AF can get close to the better position easily in relatively area. However, during the iteration process, as the step gradually gets smaller, the accuracy of solution is enhanced.

3.2.3 Inertia weight factor

In PSO, the inertia weight factor defines the impact of particle's velocity to the current one, and controls the range of the search effectively[24]. Besides, with the application of inertia weight factor, PSO algorithm can get the balance between exploration and exploitation [25]. Inspired by the PSO algorithm, we introduce inertia weight factor in MAFSA. We choose an appropriate constant for β to compromise between global exploration and local exploitation. The inertia weight factor β is determined as followed:

$$\beta = \beta_{\text{start}} - \frac{\beta_{\text{start}} - \beta_{\text{end}}}{T_{\max}} \times t \tag{16}$$

Where t denotes the current iteration number and $t_{\max}$ denotes the maximum iteration number, β_{start} and β_{end} are the inertia weight at the beginning and at the end, respectively.

3.3 New Behaviors

3.3.1 New prey behavior

This behavior is an individual behavior and each AF performs it without considering other AFs. Each AF does a local search around itself. Suppose the position of AF i is $X_i(t)$, the next position is $X_j(t)$, $X_i(t+1)$ is the position randomly generated in the visual of AF. $X_i(t+1)$ is determined as the followed:

$$X_i(t+1) = (1-\beta) X_i(t) + \beta \times Step \times Rand(-1,1) \tag{17}$$

If the food concentration of position $X_i(t+1)$ is better than the current position $X_i(t)$, this position will be replaced by

$$X_j(t) = X_i(t+1) \tag{18}$$

3.3.2 New swarm behavior

In standard AFSA, the central position of the AF swarm is calculated by equation (4), it's the arithmetic average of all the AF swarm individual positions. And the current AF i moves toward X_c with a randomly distance in the visual. In this paper, a new move strategy is introduced to improve the swarm behavior. As it's shown by the following expression:

$$\begin{aligned} X_i(t+1) = (1\text{-}\beta) X_i(t) + \beta \times [&\frac{X_C - X_i(t)}{|X_C - X_i(t)|} \times Step \times rand(0,1) \\ &+ \frac{X_{\text{best}} - X_i(t)}{|X_{\text{best}} - X_i(t)|} \times Step \times rand(0,1)] \end{aligned} \tag{19}$$

The best position of the new artificial fish swarm in each iteration is used in new swarm behavior. If the food concentration of position $X_i(t+1)$ is better than the position $X_i(t)$, the current AF i moves toward position between X_{best} and X_c with a randomly distance in the visual.

3.3.3 New follow behavior

In standard AFSA, the current AF moves toward the better AF (according to fitness value) in its visual scope, but in MAFSA, each fish moves some distance toward the position between the best AF and the better AF in its visual. The new follow behavior is performed by using following expression:

$$\begin{aligned} X_i(t+1) = (1-\beta) X_i(t) + \beta \times [&\frac{X_j - X_i(t)}{|X_j - X_i(t)|} \times Step \times rand(0,1) \\ &+ \frac{X_{best} - X_i(t)}{|X_{best} - X_i(t)|} \times Step \times rand(0,1)] \end{aligned} \tag{20}$$

3.4 The procedure of MAFSA

The modified AFSA mainly includes the following steps.

Step1: Initialize the parameters of AF: step, visual, maximum number of iterations and try_number.

Step2: Generate the opposite artificial fish and form the new AF population.

Step3: Save the optimal value in each iteration, and recorded in the bulletin.

Step4: Implementation of new prey behavior, new swarm behavior, new follow behavior.

Step5: update the optimal value in bulletin board by the better individual optimum.

Step6: If termination condition is satisfied, output the Optimization result; otherwise return to Step2.

The flowchart of MAFSA is shown as Fig. 1.

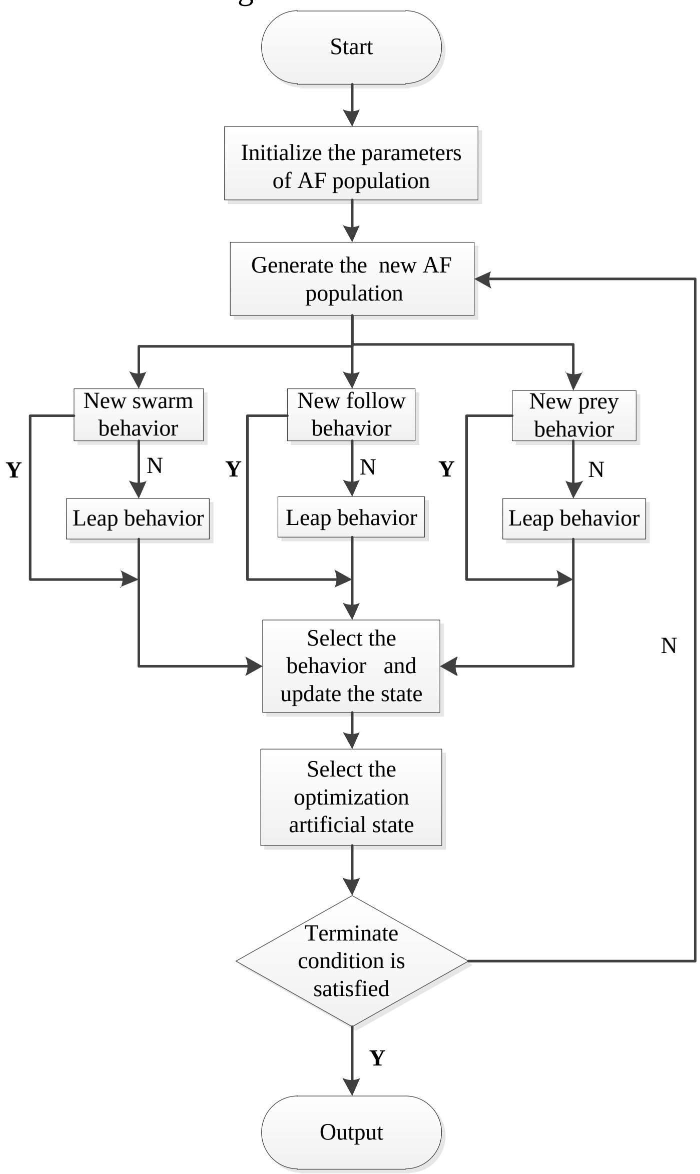


Fig. 1 Flowchart of MAFSA

3.5 Representation of environment

In general, the environment in laboratory is structural; but in nature, most of environments are un-structural autonomously. Compared with structured environment, unstructured environment is more uncertain and unstable. Thus, in simulation, we should introduce a method to deal with the unstructured environment to be structural. There are many conventional methods, such as cell decomposition[26], roadmaps[27], approximate voronoi boundary network (AVBN) [28] have been used to represent the real environment space in path planning problem. But these methods have difficulty in solving robot path planning problem in complex environments due to their low accuracy of route. To this issue, we introduce a grid-based model to represent the real environment. The method based on grid-based model is a structured that obtains the non-smooth route network by the sensor [29]. It has determined that representation of obstacle and calculation of distance is easier by using grid-based model. Also, it makes beneficial explorations for the planning under un-structural environment. As shown in Fig. 2, the environment is gridized and then each node on the grid in the obtained map is numbered sequentially.

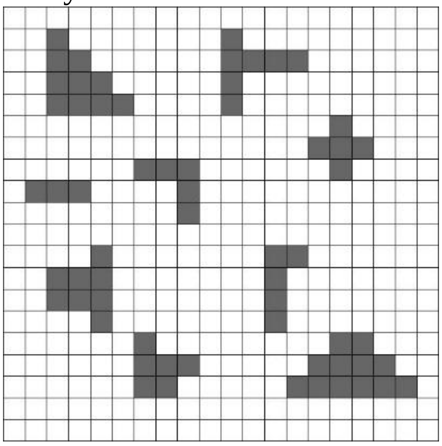


Fig. 2 Environment model

4. Optimization ability evaluation and discussion

In this section, 10 well-known benchmark functions are used as measurement to evaluate the performance of MAFSA. The Benchmark functions with their optimal value and optimal variable are shown in Table 2.

Table 2. Benchmark functions

Name	Function expression	Optimal variable	Optimal value
Rastrigin	$f(x)=\sum_{i=1}^{n}\left(x_i^2-10cos(2\pi x_i)+10\right)$	$x=(0,0,\ldots,0)$	0
Easom	$f(x)=-\cos(x_1)cos(x_2)exp\left(-(x_1-\pi)^2-(x_2-\pi)^2\right)+1$	$x=(\pi,\pi)$	0
Booth	$f(x)=(x_1+2x_2-7)^2+(2x_1+x_2-5)^2$	$x=(1,3)$	0
Bohachevsky	$f(x)=x_1^2+2x_2^2-0.3cos(3\pi x_1)-0.4cos(4\pi x_2)+0.7$	$x=(0,0)$	0
Sum Squares	$f(x)=\sum_{i=1}^{n} ix_i^2$	$x=(0,0,\ldots,0)$	0
Schaffer	$f(x)=0.5+\frac{\left[\sin\left(\sum_{i=1}^{n}x_i^2\right)\right]^2-0.5}{(1+0.001\left(\sum_{i=1}^{n}x_i^2)\right)^2}$	$x=(1,0,\ldots,0)$	0
Rosebrock	$f(x)=\sum_{i=1}^{n}\left[100(x_{i+1}-x_i)^2+(x_i-1)^2\right]$	$x=(1,1,\ldots,1)$	0
Sphere	$f(x)=\sum_{i=1}^{n}(x_i)^2$	$x=(0,0,\ldots,0)$	0
Griewank	$f(x)=\sum_{i=1}^{n}\frac{x_i^2}{4000}-\prod_{i}^{n}\left(\frac{x_i}{\sqrt{i}}\right)+1$	$x=(0,0,\ldots,0)$	0

Ackley	$f(x) = -20e^{-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}} - e^{\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)} + 20 + e$	$x = (0,0,\ldots,0)$	0

In standard AFSA and MAFSA, the initial value of visual and step has been set to 20% and 10% of the range length of the text function variables respectively. The inertia weight factor β_{start} and β_{end} are 0.3 and 0.9, respectively. The fish number is 100, crowd factor is 0.6 and the number of iteration are 100、200、300, respectively. Algorithm has been run 50 times for each iteration number, We used the mean(average optimal value) to evaluate the optimal performance of algorithms. The result is shown in Table 3.

Table 3. The average optimal value with different iteration by execute 50 times

Function Name	*Standard AFSA*			*MAFSA*		
	100 Itr	***200 Itr***	***300 Itr***	***100 Itr***	***200 Itr***	***300 Itr***
Rastrigin	0.0197416	0.0165754	0.0156553	0.0051028	0.00349	0.003035
Easom	0.021071	0.022532	0.0218015	0.003841	0.003375	0.003397
Booth	0.008165	0.0064109	0.0060774	0.000732	0.0012518	0.0010033
Bohachevsky	0.012106	0.010684	0.010507	0.003589	0.002135	0.002036
Sum Squares	0.008670	0.010458	0.010335	0.003848	0.002260	0.002003
Schaffer	0.009721	0.0079834	0.0052615	0.0010212	0.0009465	0.000891
Rosebrock	0.017311	0.01546	0.01506	0.005456	0.003135	0.003054
Sphere	0.021546	0.01965	0.018952	0.001451	0.001154	0.0001056
Griewank	0.074564	0.070301	0.007159	0.001325	0.000578	0.0005564
Ackley	0.0514564	0.0497621	0.0441546	0.001656	0.001051	0.001006

The dimensions of the Rastrigin, Sum Squares , Schaffer, Rosebrock, Sphere, Griewank, Ackley function are taken as 10, and Easom、Booth and Bohachevsky function is taken as 2 dimensions. From the table 3， we can see that the average optimal value which is gotten by modified AFSA is better than the standard AFSA. The tests show that the proposed algorithm has higher accuracy in function optimization problem, especially for multi-extreme optimization problem.

To evaluate the convergence speed of the algorithm, we run the two algorithms for about 100 times, and record the max convergence iterations number, the min convergence iterations number and the average convergence iterations number in 100 executions. The data is shown in the Table 4.

Table 4. The convergence of optimal result in 100 times experiment

Function Name	Standard AFSA			MAFSA		
	100 Itr	200 Itr	300 Itr	100 Itr	200 Itr	300 Itr
Rastrigin	135	45	85	21	8	14
Easom	27	7	19	11	5	5
Booth	24	5	17	15	4	9
Bohachevsky	34	8	25	15	2	6
Sum Squares	52	19	37	17	6	12
Schaffer	71	23	37	25	10	15
Rosebrock	98	21	48	31	13	21
Sphere	89	19	35	28	6	15
Griewank	102	29	57	38	9	23
Ackley	91	27	41	27	13	19

In table 4, we can see that with MAFSA, the max convergence iteration number、the min convergence iteration number and the average convergence iteration number of the ten text function is smaller than standard AFSA. The table shows that, compared with standard AFSA, the search ability of MAFSA is enhanced.

5. Simulation of Robot Path planning

5.1 Initialization of Population

The initial population is generally generated randomly and the generated individuals in the population are possible paths which the mobile robot may walk. Each path is represented by the equation $P=[p_1, p_2, \cdots, p_n], n \geq 2$, where p_n is the grid in the grid-based environment model. In this sturdy, the grid is the smallest unit of information in grid-based environment model and the length of the path is related to the number of grids in problem space. An optimal or near optimal path can be found by artificial fish swarm behaviors.

5.2 Population evaluation

In this section, the proposed intelligent algorithm evaluates individuals by the fitness function. The fitness function is an important factor to the convergence of MAFSA, which considers the total slope of the path and the length, it take shorter time to pass though the region contains obstacle when the path is shorter. Here, the fitness function is the total length of the path and it is used to define paths which are determined by the sum of distances. The fitness function is given by the follow equation:

$$f=\begin{cases}\sum_{i=1}^{n-1} D(p_i, p_{i+1}) & \text{feasible} \\ \sum_{i=1}^{n-1} D(p_i, p_{i+1}) + addition & \text{infeasible}\end{cases} \tag{21}$$

Where $D(p_i p_{i+1})$ is the Euclidean distance between the position p_i and position p_{i+1}, it is given by:

$$D(p_i p_{i+1}) = \sqrt{(x_{i+1}-x_i)^2 + (y_{i+1}-y_i)^2} \tag{22}$$

In general, the obtained individuals contain infeasible and feasible paths. So, in evaluation mechanism, different fitness function is built for the feasible and infeasible individual in the population, respectively. In equation (21), a restriction mechanism is introduced in the fitness function for infeasible path. If there are some obstacles intersect along the path of the robot, an addition value is added to fitness function to make sure that the value of any given infeasible path is worse than those of feasible paths. The addition value is greater than the maximum path length the grid-based environment model. To seek an optimal path, the algorithm searches for an individual whose addition is eliminated.

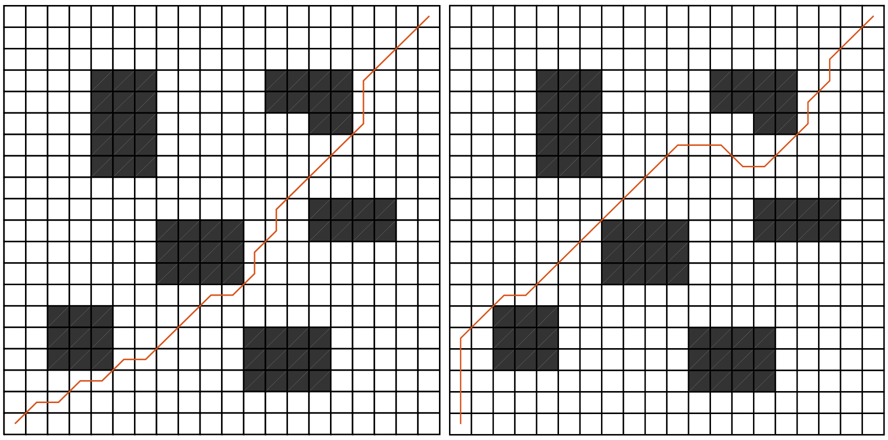
5.3 Simulation on grid-based model

In this section, the simulation of the proposed algorithm applied in robot path planning is conduct in MATALB 2009a. The approach is validated in two different environments: the simple environment and the complex environment. And the simulation is compared with previous algorithm in literature. Each environment contains series of irregular and regular obstacles, which make these environments much more complicated. In robot path planning, the length of the path is usually selected as the optimization criterion, and the given solution is optimal or near optimal. The first environment simulates simple scenario where there are five obstacles, and the last environment has 11 obstacles. Compared with the simple environment, the number of obstacles in complex environment is higher.

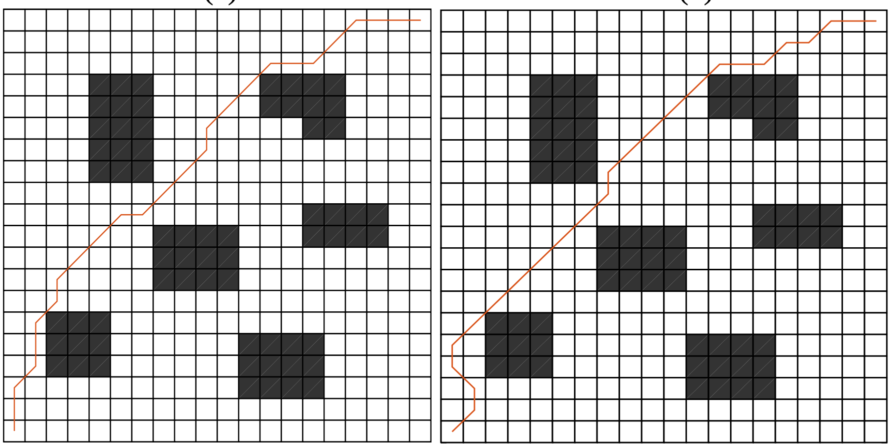
In generally, it is a critical aspect to set parameters in MAFSA. But, there is no formal way to identify the appropriate parameters. In the traditional approach, this is done experimentally. Also, in this paper, the parameters are obtained after several tests. In the following experiments, the proposed algorithm is run using the parameter as follows: the population is set to size N=50,the maximum number of alterations $T_{\max}=300$, the initial value of visual and step has been considered 30% and 10% of the range length of the fitness function variables respectively. The inertia weight factor β_{start} and β_{end} are 0.4 and 0.8. Algorithm has been run 50 times for each iteration number, We used the mean(average optimal value) to evaluate the optimal performance of algorithms. The result is shown in the Table 2.

5.3.1. Environment 1

First, the proposed algorithm is tested in a simple environment which consists of 20×20 grids. The simple environment contains six obstacle regions in black color, white grids represent obstacle-free nodes and gray grids represent obstacle nodes. The trajectories of the robot in given environment are represented by simple red lines. Simulation result of path planning is given in Fig.3, Figure 4(a),(b),(c),(d) correspond to the optimal path computed by the MAFSA, HAAFSA, IDAFSA, IAFSA, respectively. It can be observed from the results that all the above-mentioned algorithms can finish the robot path planning and enable the robot avoiding the obstacles. Also, it should be noted that the self-adaptive strategy based on opposition-based reinforcement learning enables the robot locate the area with the target point in a shorter route. That is, the self-adaptive strategy can efficiently find the optimal path and actuators' energy of the mobile robot.



(a) (b)

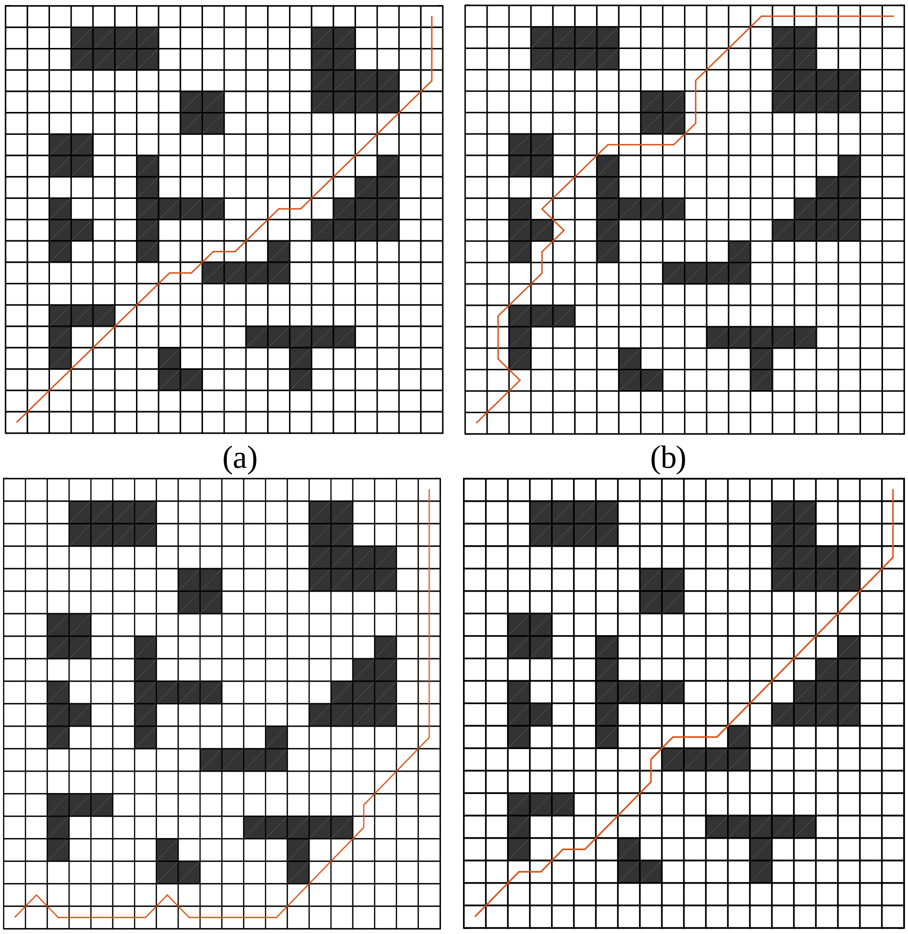


(c) (b)

Fig. 3 The simulation in simple environment

5.3.2. Environment 2

In this simulation, a more complex environment which consists of 20×20 grids and has eleven obstacles regions are used to test the performance of proposed algorithms, white grids represent obstacle-free nodes and gray grids represent obstacle nodes. As shown in Fig.4, more regular and irregular obstacles are added. The trajectories of the robot are represented by simple red lines. Simulation result of path planning is given in Fig.5, Figure 5(a),(b),(c),(d) correspond to the optimal path computed by the MAFSA, HAAFSA, IDAFSA, IAFSA, respectively. From the Figure we can that the mentioned methods can solve robot path planning problem. But, we should note that the route which is got by proposed MAFSA based on opposition-based reinforcement learning in a shortest. The proposed MAFSA has powerful search ability and optimal ability in mobile robot path planning.



(a) (b)

(c) (b)

Fig. 4 The simulation in complex environment

5.4 Analysis of simulation result

The simulation results in the simple and complex environments are given Table5 and Table 6, respectively. As we know, the more obstacles cause more environmental complexity, while the environment is more complicated, the experimental results given by intelligent algorithms are not identical each time. However, all the results are obtained by intelligent algorithms are optimal or near-optimal. In Table 5, Table contains the average value of the route, the Maximum value of the route and the minimum value of the route and the standard deviation in 50 trials. It is clearly seen that the MAFSA based on opposition-based reinforcement learning has powerful ability in finds the optimal path. The average value, the Maximum value and the minimum value of the rout is the lowest. And the standard deviation is also small. The average fitness values of the proposed method are better than the other methods' values.

Table 5. The Result of experiment for the two environments

Environment	Algorithm	Mean	Best	Worst
environment 1	HAFSA	32.4532	30.4385	34.8632
	IDAFSA	33.0352	31.1832	39.5453
	AFSA	32.9482	30.2314	35.7667
	MAFSA	30.0743	29.7991	32.0146
environment 2	HAFSA	31.6752	29.1212	36.8654
	IDAFSA	32.1676	29.4123	37.7536
	AFSA	32.3512	29.1324	35.5564
	MAFSA	29.2652	28.6725	30.6274

The number of near optimal solutions, the optimal solution and the Success rate found in 50 trials are given in Table 6, It can be seen that MAFSA can find the best path for about 41 and 39 times in environment 1 and environment 2 respectively, while the other methods find it only several times. The experimental result indicates that it has high stability and effectiveness in path planning. The higher Success rate also shows that the proposed method can get better performance than the other methods in path planning.

Table 6. The Result of experiment for the two environments

Environment	*Algorithm*	*Total solution*	*Optimal solution*	*Near optimal solution*	*Success rate*
environment 1	HAFSA	50	29	21	58%
	IDAFSA	50	27	23	54%
	AFSA	50	11	39	22%
	MAFSA	50	41	9	82%
environment 2	HAFSA	50	19	31	38%
	IDAFSA	50	21	29	42%
	AFSA	50	9	41	18%
	MAFSA	50	39	11	78%

6. Conclusion

In this paper, MAFSA, a new approach to deal with robot path planning in complex environments is proposed. The approach uses a new modified opposition-based reinforcement learning mechanism to update the population of standard AFSA. To address the shortcoming of visual in standard AFSA, an attenuation function α is introduced. Also, an adaptive operator is introduced to enhance the adaptive of step. Besides, we introduce a self-active inertia weight factor to improve the convergence. The MAFSA is applied to the path planning problem of mobile robot which is installed ROS. Ten well-known benchmark functions are used to test the performance of MAFSA. The test result shows that the convergence speed of MAFSA is more efficient, the local search ability is enhanced and the optimal result is more accurate. Simulation of path planning is performed on grid-based model, the results show that MAFSA has strong optimization ability in robot navigation.

The future work will be done as follows. First, we will extend the model in a highly complicated environment to evaluate the self-adaptive ability, and adds practical applicability. Second, we are interested in using MAFSA for robot path planning in complex and dynamic environments.

Acknowledgements

This work was supported by the Chongqing Science and Technology Commission (CSTC) (no. cstc 2015jcyjBX0066).

References

[1] I. A. Taharwa, A. Sheta and M. A. Weshah: A mobile robot path planning using genetic algorithm in static environment, Journal of Computer Science, Vol. 4 (2008) No. 4, p. 341-344.

[2] X. P. Wei, J. X. Zhang, D. S. Zhou and Q. Zhang: Optimal path planning for minimizing base disturbance of space robot, International Journal of Advanced Robotic Systems, Vol. 13(2016) No. 41.

[3] H. Mo and L. Xu: Research of biogeography particle swarm optimization for robot path planning, Neurocomputing, Vol. 148(2015) No. 148, p. 91-99.

[4] Y. Hu, S. X. Yang, L. Z. Xu and M. Q. H. Meng: A Knowledge Based Genetic Algorithm for Path Planning in Unstructured Mobile Robot Environments, *IEEE International Conference on Robotics and Biomimetics* (Shenyang, China, August 22-26, 2004).

[5] X. Hu and Q. Xu: Robot path planning based on artificial immune network, *IEEE International Conference on Robotics and Biomimetics* (Sanya, China, December 15-18, 2007).

[6] I. Hassanzadeh, H. Ghadiri and R. Dalayimilan: Design and implemention of a simple fuzzy algorithm for obstacle avoidance navigation of a mobile robot in dynamic environment, International Symposium on Mechatronics and Its Applications, (2008), p.1-6.

[7] Y. Zhang, P. Agarwal, V. Bhatnagar, S. Balochian, and X. Zhang: Swarm intelligence and its applications 2014, Scientific World Journal, Vol. 2013(2013) No. 5.

[8] R. Song, X. Chen, C. Shen and H. Zhang: Modeling fog drift using back-propagation neural network optimized by artificial fish swarm algorithm, Journal of Sensors, (2014), p.1-6.

[9] X. L. Li., Z. J. Shao and J. X. Qian: An optimizing method based on autonomous animats: fish-swarm algorithm, Systems Engineering-theory and Practice, Vol. 22 (2002) No. 11, p. 32-38.

[10] Z. Ni: An improved discrete optimization algorithm based on artificial fish swarm and its application for attribute reduction, Journal of Information and Computational Science, Vol. 12 (2015) No. 6, p. 2143-2154.

[11] R. Azizi: Empirical study of artificial fish swarm algorithm, Computer Science, Vol. 17(2014) No. 6, p. 626-641.

[12] P. Li, X. Huang and M. Wang: A new hybrid method for mobile robot dynamic local path planning in unknown environment, Journal of Computers, Vol. 5(2010) No. 5, p. 773-781.

[13] A. Tuncer and M. Yildirim: Dynamic path planning of mobile robots with improved genetic algorithm, Computers and Electrical Engineering, Vol. 38(2012) No. 38, p. 1564-1572.

[14] J. S. Pan and P. Shi: Special issue: swarm intelligence and its applications, Information Sciences, Vol. 192(2012) No. 6, p. 1-2.

[15] Q. Ma and X. Lei: Application of artificial fish school algorithm in UCAV path planning, *Proceedings of the 15th International Conference on Bio-Inspired Computing: Theories and Applications* (Changsha, China, September 23-26, 2010).

[16] J. Peng, X. Li, Z. Q. Qin and G. Luo: Robot global path planning based on improved artificial fish-swarm algorithm, Research Journal of Applied Sciences Engineering and Technology, Vol. 5 (2013) No. 6, p. 2042-2047.

[17] Z. Yao and Z. Ren: Path planning for coalmine rescue robot based on hybrid adaptive artificial fish swarm algorithm, International Journal of Control and Automation, Vol. 7(2014) No. 8, p. 1-12.

[18] X. Ma: Application of adaptive hybrid sequences niche artificial fish swarm algorithm in vehicle routing problem, *Proceedings of the 2nd International Conference on Future Computer and Communication* (Wuhan, China, May 21-24, 2010).

[19] Z. Ni: An improved discrete optimization algorithm based on artificial fish swarm and its application for attribute reduction, Journal of Information and Computational Science, Vol. 12 (2015) No. 6, p. 2143-2154.

[20] D. Lu, G. Zhang, Y. Liu, D. Wang and H. Liu: Afsa based path planning method for crowd evacuation, Journal of Information and Computational Science, Vol. 11(2014) No. 11, p. 3815-3823.

[21] Y. Lei, and Z. Feng: The Optimization of Fuzzy Neural Network Based on Artificial Fish Swarm Algorithm, *Proceedings of the 9th International Conference on Mobile Ad-Hoc and Sensor Networks* (Dalian, China, December 11-13, 2013).

[22] D. Yazdani, B. Nasiri, A. Sepas-Moghaddam, M. Meybodi, and M. Akbarzadeh-Totonchi: Mnafsa: a novel approach for optimization in dynamic environments with global changes, Swarm and Evolutionary Computation, Vol. 18(2014) No. 1, p. 38-53.

[23] L. Chen and X. Zhao: An improved power control afsa for minimum interference to primary users in cognitive radio networks, Wireless Personal Communications, Vol. 87(2016) No. 1, p. 293-311.

[24] G. Chen, Z. Min, J. Jia and X. Huang: Self-Active Inertia Weight Strategy in Particle Swarm Optimization Algorithm, *Proceedings of the 6th World Congress on Intelligent Control and Automation* (Dalian, China, January, 2006).

[25] K. Kentzoglanakis and M. Poole: Particle swarm optimization with an oscillating inertia weight, *Proceedings of the 11th Annual conference on Genetic and Evolutionary Computation Conference* (Montreal, Canada, July 8-12, 2009).
[26] C. Cai and S. Ferrari: Information-driven sensor path planning by approximate cell decomposition, IEEE Transactions on Systems Man & Cybernetic*s*, Vol. 39(2009) No. 3, p. 672-689.
[27] M. J. Anderson, T. O. Crist, J. M. Chase, M. Vellend, B. D. Inouye and A. L. Freestone: Navigating the multiple meanings of β diversity: a roadmap for the practicing ecologist, Ecology Letters, Vol. 14 (2011) No. 1, p. 19-28.
[28] H. Mo and L. Xu: Research of biogeography particle swarm optimization for robot path planning, Neurocomputing, Vol. 148(2015) No. 148, p. 91-99.
[29] W. Tao and M. Chen: Data processing and algorithm analysis of vehicle path planning based on wireless sensor network, International Journal of Distributed Sensor Networks, Vol. 2013(2013) No. 3, p. 179-180.