

Research Article

The Robot Path Planning Based on Improved Artificial Fish Swarm Algorithm

Yi Zhang,¹ Guolun Guan,¹ and Xingchen Pu²

¹Research Center of Intelligent System and Robot, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

²School of Science, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

Correspondence should be addressed to Guolun Guan; jianyoutianxia@163.com

Received 19 February 2016; Accepted 1 August 2016

Academic Editor: Fei Liu

Copyright © 2016 Yi Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Path planning is critical to the efficiency and fidelity of robot navigation. The solution of robot path planning is to seek a collision-free and the shortest path from the start node to target node. In this paper, we propose a new improved artificial fish swarm algorithm (IAFSA) to process the mobile robot path planning problem in a real environment. In IAFSA, 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 introduced to enhance the adaptive ability of step. Besides, a concept of inertia weight factor is proposed in IAFSA inspired by PSO intelligence algorithm to improve the convergence rate and accuracy of IAFSA. Five unconstrained optimization test functions are given to illustrate the strong searching ability and ideal convergence of IAFSA. Finally, the ROS (robot operation system) based experiment is carried out on a Pioneer 3-DX mobile robot; the experiment results also show the superiority of IAFSA.

1. Introduction

With the popularity and wide application of mobile robots in the engineering field, the issue of path planning is going to be a spotlight in the navigation of mobile robots in the future. The goal of path planning is to find an optimal collision-free and the shortest path from a start node to a target node in a given real environment [1, 2]. The optimal collision-free path is usually measured by some criterions, such as distance, time, or energy; distance and time are the most commonly adopted criterions [3]. Hence, a lot of algorithms are applied to solve the problem of path planning, such as genetic algorithm, ant colony algorithm, PSO, neural networks, and fuzzy logic [4, 5]. Artificial fish swarm algorithm (AFSA) [6], a branch of Swarm Intelligence, was firstly brought forward by Li et al. in 2002. Through simulating the social behavior of fish in nature, the algorithm is widely used to process the optimization problem [7–9].

Recently, with its strong capacity of global search, fast convergence rate, and good robustness, AFSA is used to deal with robot path planning problem [10]. Therefore, plenty of researchers made some attempts to improve the performance

of AFSA. In [11], 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 ambient link of fish foraging behavior. By examining the environment, artificial fish can screen the information from the surroundings to obtain an optimal state for the best direction of movement. Yao and Ren 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; the HAAFSA was applied to coalmine rescue robot path planning and the performance of the optimal path is improved [12]. In [13], the adaptive hybrid sequences niche artificial fish swarm algorithm (AHSN-AFSA) is presented to solve the vehicle routing problem, and the concept of ecological niche is also proposed to improve the shortcoming of traditional AFSA to obtain an optimal solution. In [14], a new idea based on least squares support vector machine (LSSVM), whose parameters are optimized by artificial fish swarm algorithm (AFSA), has been used to improve location accuracy of three-dimensional sensor nodes of wireless sensor network. The experimental

result shows that, compared with other localization methods, the proposed method can improve the accuracy of sensor nodes. A modified version of artificial fish swarm algorithm (MAFSA) with a mechanism to dynamically control the parameter vision is also being introduced to select the optimal feature subset to improve the classification accuracy for support vector machines; experimental results show that the performance of the proposed method is superior to standard AFSA [15]. In [16], a novel AFSA algorithm, so-called NAFSA, has been proposed to improve weak points of standard AFSA and accelerate convergence velocity of the algorithm. Also, a multiswarm algorithm based on NAFSA was introduced to conquer particularly the dynamic environment by proposing some new mechanism including multiswarm mechanism for finding and diversity increase mechanism and covering potential optimum peaks. Extensive experiments show that the proposed algorithm can get better performance for optimization in dynamic environments.

All the above algorithms are successful in improving the weak point of standard AFSA and get better performance in the corresponding fields. Despite the fact that the mentioned methods can improve the performance in some cases, they still cannot reach the ideal result when it comes to some special optimization problems. In this paper, a new improved artificial fish swarm algorithm (IAFSA) is proposed and applied in the path planning problem. In IAFSA, an attenuation function α is introduced to improve the visual. Also, an adaptive operator based on Gaussian distribution function is introduced to enhance the adaptive ability of step. Besides, inspired by PSO algorithm, an inertia weight factor is introduced in IAFSA to improve the fish behaviors and balance the exploration and exploitation. Simulation shows that IAFSA is more effective, especially when it is used to deal with multipeak and large search space function optimization problem.

The remainder of this paper is organized as follows. Section 2 introduces the standard artificial fish swarm algorithms. In Section 3, the proposed method is presented. Section 4 shows the simulation cases and analyzes the results. Section 5 studies the experiment in mobile robot and analyzes the result. Finally, conclusion and discussion are given in Section 6.

2. Standard AFSA

Artificial fish (AF) is a fictitious entity of true fish. Generally, fish moves to a position with better food consistency by performing social search behaviors [10]. AF has approximately four social behaviors: prey behavior, follow behavior, swarm behavior, and leap behavior [17]. We use these behaviors to conduct the analysis and explanation of the problem.

Suppose the state of individual artificial fish is vector X . We can denote the vector $X = (x_1, x_2, \dots, x_n)$, where x_i ($i = 1, 2, \dots, n$) is the optimization variable of fishes. The food concentration in the current position of the artificial fish can be expressed as $Y = f(X)$, 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 following equation:

$$D_{i,j} = |X_i - X_j|. \quad (1)$$

The crowd factor δ ($0 < \delta < 1$) 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.

(1) *Prey Behavior.* In nature, prey behavior is a basic biological behavior for 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 , 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 following equation:

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

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

$$\begin{aligned} X_i(t+1) &= X_i(t) + \frac{X_j(t) - X_i(t)}{\|X_j(t) - X_i(t)\|} \times \text{Step} \\ &\quad \times \text{rand}(0, 1). \end{aligned} \quad (3)$$

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

(2) *Swarm Behavior.* In order to keep swarm generality, AFs attempt to move towards the center position in every time of iterations. The central position can be determined as the following equation:

$$X_c = \frac{1}{N} \sum_1^N X_i, \quad (4)$$

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 swarms 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 better food consistence and population is not a crowd, AF i moves forward a step to the companion center by

$$\begin{aligned} X_i(t+1) &= X_i(t) + \frac{X_c - X_i(t)}{\|X_c - X_i(t)\|} \times \text{Step} \\ &\quad \times \text{rand}(0, 1). \end{aligned} \quad (5)$$

Otherwise, AF performs prey behavior.

(3) *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 swarms in the visual scope of X_c ; if $Y_i < Y_j$ and $n_f/n < \delta$, AF i moves forward a step to the neighbor X_j . The expression is determined as follows:

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

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

(4) *Leap Behavior.* Leap behavior is the basic behavior to seek food or companions in large ranges, which can effectively prevent local optimum. AF i performs the leap behavior and changes the parameter to leap out of the current position. It chooses a state in the visual and moves towards this state to avoid the local extreme values. Hence,

$$X_i(t+1) = X_i(t) + \text{Visual} \times \text{rand}(0, 1). \quad (7)$$

3. Improved AFSA

In standard AFSA, the visual and step are fixed. The visual determines the search scope, while the step determines the convergence rate and accuracy of standard AFSA. At the initial state of optimization process of standard AFSA, the large values of visual and step lead the AFs to quickly move close to the global optimal solution and improve the convergence speed. However, after the AFs move close to the target position, the large values of visual and step will make the AFs miss the position with better food concentration. Conversely, if the values of visual and step are too small, the local search ability is increased, while the global search ability is decreased. As a result, AFs may pass the global optimal solution and fall into local optimum; the accuracy of the result is decreased. In order to improve those shortcomings of standard AFSA, we proposed a new algorithm with some improved parameters.

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 once, shortening exploration and expediting convergence; it is widely applied 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.

Consider a and b as the upper and lower limits of problem space; suppose the current position of AF i is X_i ($X_i \in [a, b]$, $i \in [1, n]$), the opposite position of X_i is defined as \tilde{X}_i , and \tilde{X}_i can be calculated by the following equation:

$$\tilde{X}_i = a + b - X_i \quad i \in [1, n]. \quad (8)$$

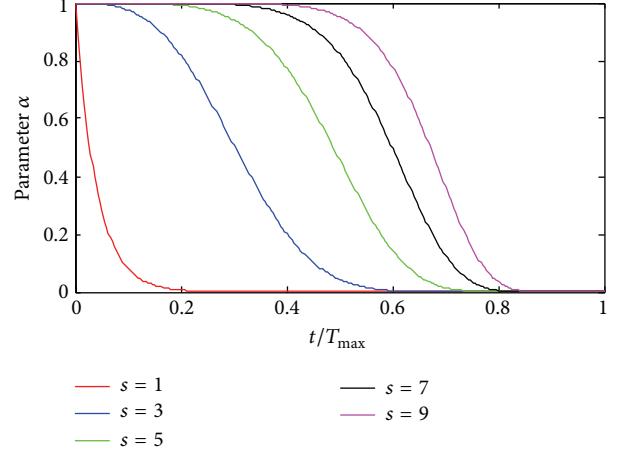


FIGURE 1: The varying graph of parameter α .

We can get the opposite position of each artificial fish and construct the opposite artificial fish swarm by (8). The food concentrations of X_i and \tilde{X}_i are $Y(X_i)$ and $Y(\tilde{X}_i)$. If $Y(X_i) > Y(\tilde{X}_i)$, we choose X_i as the new individual; otherwise, we choose \tilde{X}_i . The process is described as follows:

$$X_i = \begin{cases} X_i & Y(X_i) > Y(\tilde{X}_i) \\ \tilde{X}_i & Y(X_i) < Y(\tilde{X}_i) \end{cases} \quad i \in [1, n]. \quad (9)$$

After the above process is completed, the new artificial fish swarm, which is constituted with new individuals, is generated.

3.2. Parameter Variation on IAFSA

(1) *Visual.* In order to satisfy the requirement of convergence rate and the balance of global search and local search, a parameter α is introduced to improve the visual of AF. The parameter α is determined as follows:

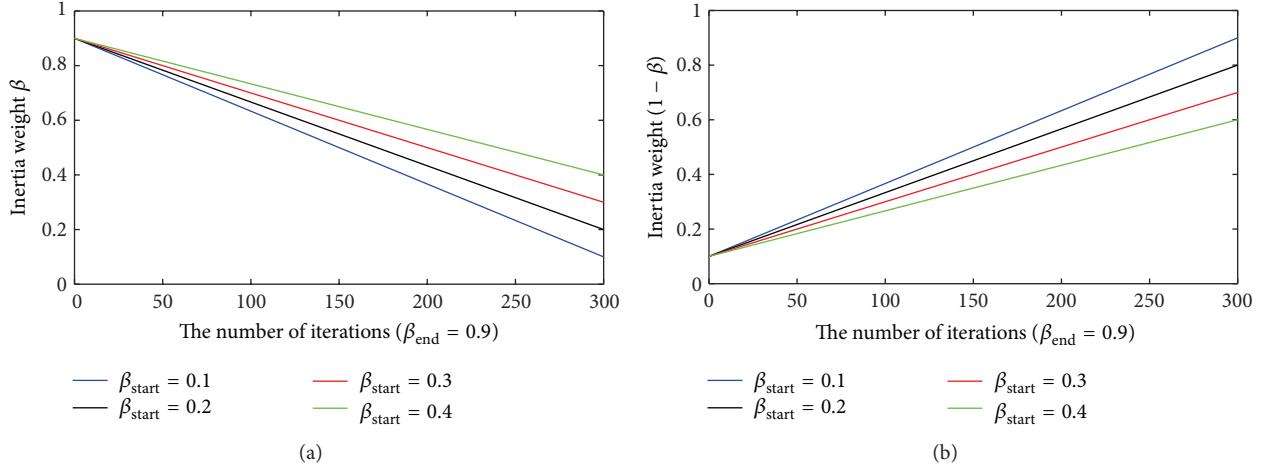
$$\alpha = \exp\left(-25 \times \left(\frac{t}{T_{\max}}\right)^s\right), \quad (10)$$

where t denotes the current iteration number, T_{\max} denotes the maximum iteration number, and the range of values for parameter s is 1 to 20. α is an attenuation function which can minimize the value of α while AF moves close to the target position during the iteration process. In Figure 1, the varying graph of parameter α in different value of s is displayed. In the early stage of iteration process, if the value of s is small, the attenuation velocity of α is quick; conversely, if the value of s is too large, the attenuation of α is slow. Thus, in IAFSA, the value of s is set to 3.

The improved visual is determined as follows:

$$\text{visual} = \text{visual} \times \alpha + \text{visual}_{\min}, \quad (11)$$

where visual_{\min} is the minimum determined by the requirement of the problem space. In the initial stage of the iteration

FIGURE 2: The varying graph of parameters β and $(1 - \beta)$.

process, the value of visual is relatively large and the global search ability is great; AFs can move to better positions easily in relative area. While the value of visual is gradually becoming smaller, the global search ability is decreased and local search ability is enhanced.

(2) *Step*. To balance the accuracy of solution and iteration speed, the Gaussian distribution function is introduced to improve the step of AF. The expression of Gaussian distribution function is determined as follows:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma^2}, \quad (12)$$

where μ denotes the mathematical expectation and σ denotes the standard deviation. Here, we set μ to 0. We introduce Gaussian distribution function as the adaptive operator to enhance the adaptive step. The expression of adaptive step is determined as follows:

$$\text{Step} = \text{Step} \times f\left(\frac{t}{T_{\max}}\right) + \text{Step}_{\min}, \quad (13)$$

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

(3) *Inertia Weight Factor*. In standard AFSA, AF moves to the next position by performing swarm behavior, follow behavior, leap behavior, or prey behavior. All the social behaviors play an important role in the process of optimization, especially the swarm behavior and follow behavior. In the above two behaviors, each AF moves to the next position based on a random percentage of step, from the last position in each iteration. As shown in (3) and (5), the last position and random step make a big effect in the moving process. In

the early stage of optimization, the last position makes the global search ability decrease; but at the end of optimization, the random step makes the local search ability decrease. Thus, to a certain extent, in this search mechanism, the moving flexibility of AFs is weak. Consequently, AFs cannot get the balance between global search and local search abilities. To increase the moving flexibility and balance the global search and local search ability of AFs, the inertia weight factor, which is inspired by the PSO algorithm, is introduced in IAFSA. In PSO, the inertia weight factor defines the impact of particle's velocity on the current one and controls the range of the search effectively [18]. Besides, with the application of inertia weight factor, PSO algorithm can get the balance between exploration and exploitation [19]. Here, we choose an appropriate constant for β to compromise between global exploration and local exploitation in IAFSA. The inertia weight factor β is determined as follows:

$$\beta = \beta_{\text{start}} - \frac{\beta_{\text{start}} - \beta_{\text{end}}}{T_{\max}} \times t, \quad (14)$$

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. In Figure 2, the varying graphs of parameters β and $(1 - \beta)$ in different values of β_{start} and β_{end} are displayed. In the early stage of the iteration process, the value of β is large and $(1 - \beta)$ is small; the global search ability is enhanced. And, during the process of iteration, the value of β is gradually decreasing and $(1 - \beta)$ is gradually increasing; the local search ability is enhanced. Therefore, in IAFSA, AFs can get the balance between global search and local search abilities.

3.3. New Behaviors

(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 $X_i(t)$ and $X_i(t+1)$ is the position randomly generated

TABLE 1: Attributes of Cleveland dataset.

Name	Function expression	Optimal variable	Optimal value
Rastrigin	$f(x) = \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i) + 10)$	$x = (0, 0, \dots, 0)$	0
Easom	$f(x) = -\cos(x_1) \cos(x_2) \exp(-(x_1 - \pi)^2 - (x_2 - \pi)^2) + 1$	$x = (\pi, \pi)$	0
Booth	$f(x) = (x_1 + 2x_2 - 7)^2 + (2x_1 + x_2 - 5)^2$	$x = (1, 3)$	0
Bohachevsky	$f(x) = x_1^2 + 2x_2^2 - 0.3 \cos(3\pi x_1) - 0.4 \cos(4\pi x_2) + 0.7$	$x = (0, 0)$	0
Sum Squares	$f(x) = \sum_{i=1}^n i x_i^2$	$x = (0, 0, \dots, 0)$	0

in the visual of AF. The position $X_i(t+1)$ is determined as follows:

$$X_i(t+1) = (1 - \beta) X_i(t) + \beta \times \text{Step} \times \text{rand}(-1, 1). \quad (15)$$

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

$$X_i(t+1) = X_i(t). \quad (16)$$

(2) *New Swarm Behavior.* In the new swarm behavior, the central position of the AF swarm is calculated by (4); it is the arithmetic average of all the AF individual positions. The next position $X_i(t+1)$ is obtained by

$$X_i(t+1) = (1 - \beta) X_i(t) + \beta \times \frac{X_C - X_i(t)}{|X_C - X_i(t)|} \times \text{Step} \times \text{rand}(0, 1). \quad (17)$$

If the food concentration of position $X_i(t+1)$ is better than of position $X_i(t)$, the current AF i moves towards X_C with a random distance in the visual.

(3) *New Follow Behavior.* In standard AFSA, the current AF moves towards the better AF (according to fitness value) in its visual scope, but, in IAFSA, each fish moves some distance towards the best AF in the swarm by the following expression:

$$X_i(t+1) = (1 - \beta) X_i(t) + \beta \times \frac{X_{\text{best}} - X_i(t)}{|X_{\text{best}} - X_i(t)|} \times \text{Step} \times \text{rand}(0, 1). \quad (18)$$

3.4. *The Procedure of IAFSA.* The improved AFSA mainly includes the following steps.

Step 1. Initialize the parameters of artificial fish, such as step, visual, try_number, and maximum number of iterations.

Step 2. Construct the opposite artificial fish and form the new artificial fish population by (8) and (9).

Step 3. Select the optimal value of each fish, and record it in the bulletin.

Step 4. Implement prey behavior, swarm behavior, follow behavior, and leap behavior.

Step 5. The optimal value in the bulletin board is updated by the better individual optimum.

Step 6. If termination condition is satisfied, output the optimization result; otherwise, return to Step 2.

The flow chart of IAFSA is shown in Figure 3.

4. Simulation and Discussion

In this section, five unconstrained benchmark functions are used as measurement to evaluate the performance of IAFSA. The benchmark functions with their optimal value and optimal variable are shown in Table 1.

Easom, Booth, and Rastrigin functions are multipeak functions with several local minimums, and Sum Squares and Bohachevsky functions are single-peak functions which have no local minimum except the global one. We can find the global minimum by analysis for the function expression.

In standard AFSA and IAFSA, the initial value of visual and step has been considered to be 30% and 10% of the range length of the fitness function variables, respectively. The inertia weight factors β_{start} and β_{end} are 0.2 and 0.9. The fish number is 100, crowd factor is 0.6, and the numbers of iterations are 50, 100, and 200, respectively. The algorithm is run 50 times for each iteration number. Here, we used the mean (average optimal value) to evaluate the optimal performance of the algorithms. The result is shown in Table 2.

The dimensions of the Rastrigin function and Sum Squares function are taken as 10, and those of Easom, Booth, and Bohachevsky functions are taken as 2. From Table 2, we can see that the average optimal value which is got by IAFSA is better than the standard AFSA. The tests show that the proposed algorithm has higher accuracy in function optimization problem, especially for multiextreme optimization problem.

To evaluate the convergence speed of the algorithm, the two algorithms are run about 100 times and record the max convergence iteration number, the min convergence iteration number, and the average convergence iteration number. The data is shown in Table 3. And the optimization process is shown in Figure 2.

In Table 3, we can see that, with the improved AFSA, the max convergence iteration number, the min convergence iteration number, and the average convergence iteration number of the five test functions are smaller than standard

TABLE 2: The average optimal value with different iterations.

Function name	Standard AFSA			Improved AFSA		
	50 Itr	100 Itr	200 Itr	50 Itr	100 Itr	200 Itr
Rastrigin	0.019541667	0.016765714	0.01565533	0.004902857	0.00363	0.002735
Easom	0.021071	0.022532	0.0218015	0.0030341	0.003865	0.00344955
Booth	0.007065	0.005180909	0.006078095	0.000732	0.001251818	0.001003333
Bohachevsky	0.010106	0.013684	0.008507	0.002489	0.003235	0.002446
Sum Squares	0.008670	0.011748	0.011235	0.001848	0.003260	0.004433

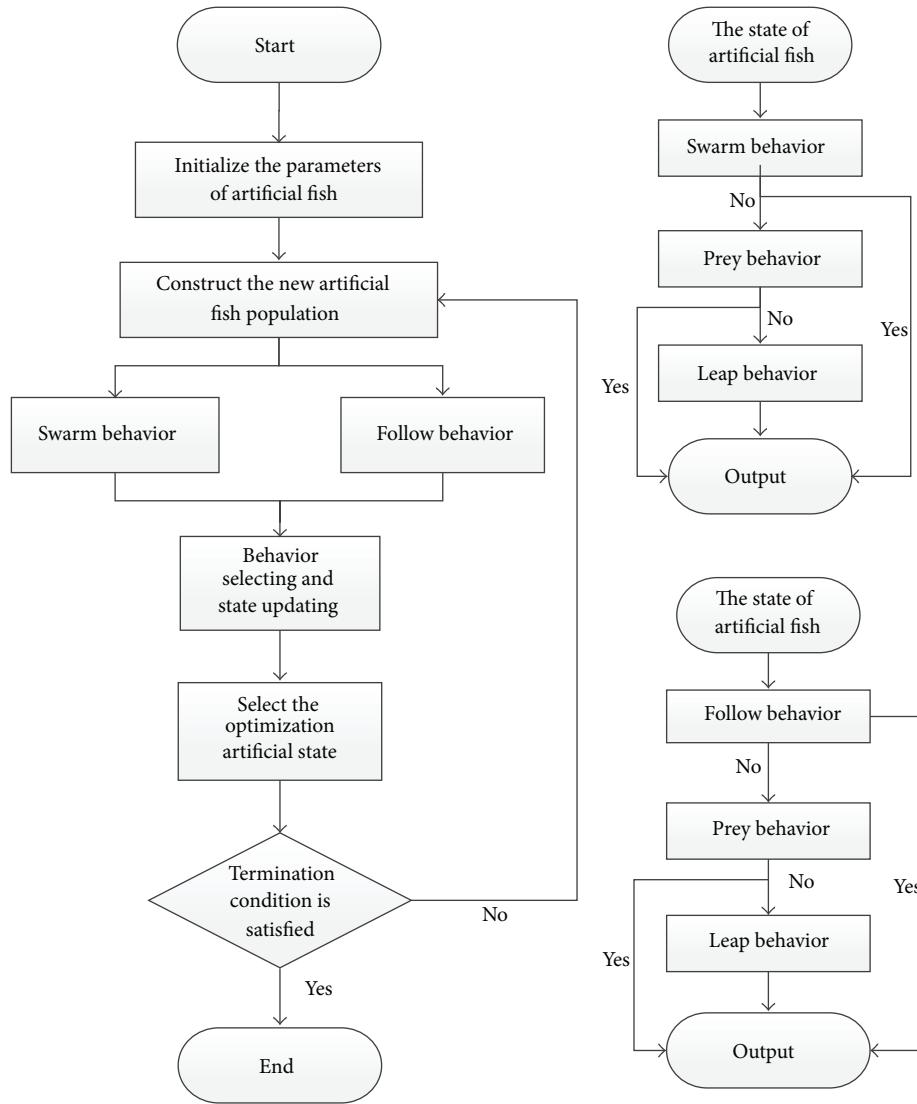


FIGURE 3: The flow chart of IAFSA.

AFSA. The table shows that, compared with standard AFSA, the search ability of IAFSA is enhanced.

As shown in Figure 4, the search ability of IAFSA is stronger than of AFSA, and the convergence speed is faster. IAFSA is shown to have better performance in improving search accuracy and speed; also, to a certain extent, the searching efficiency, global optimization ability, and accuracy of the solution are improved.

5. Experiment and Discussion

5.1. Introduction of Experiment Platform. The experiment in a real environment is further investigated on the platform of Pioneer 3-DX (Figure 5) mobile robot which is equipped with ROS (robot operation system). With its modest and balanced size combined with reasonable hardware, the mobile robot is suitable for indoor laboratory or classroom navigation.

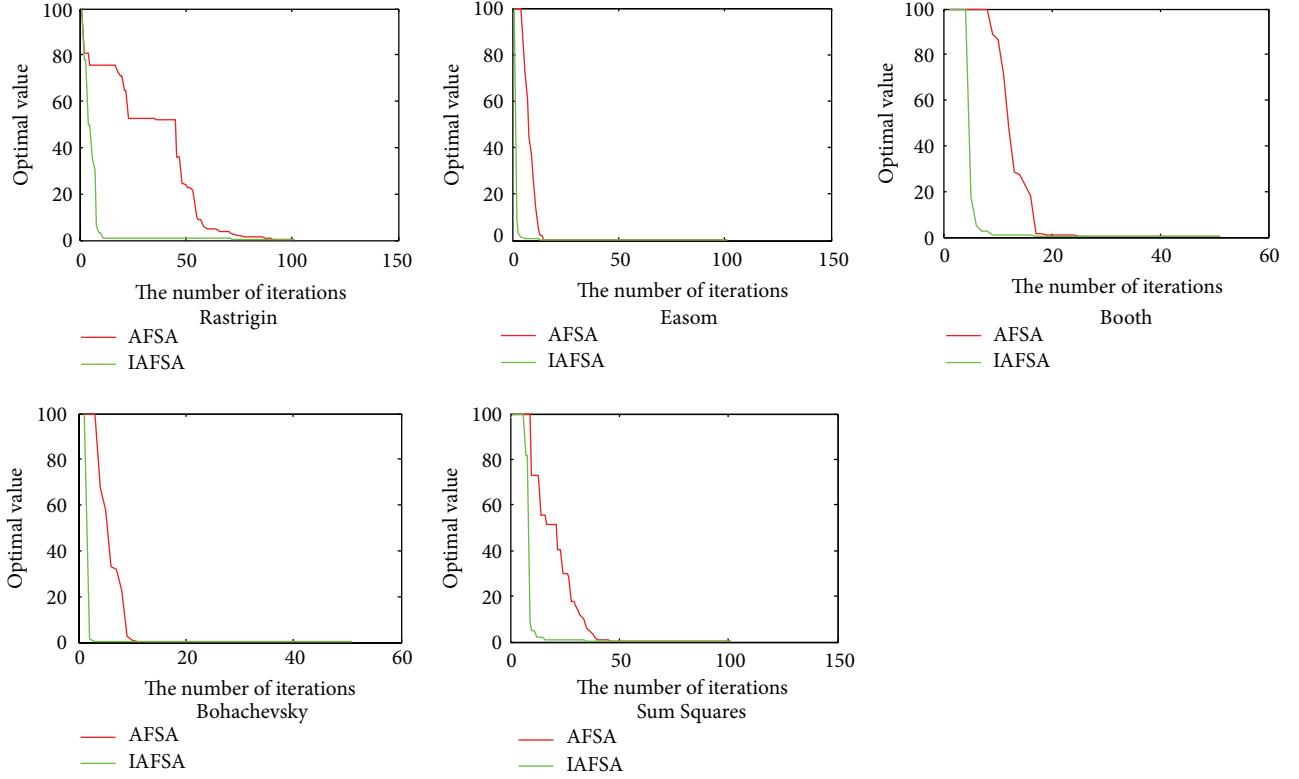


FIGURE 4: The optimal process.

TABLE 3: The convergence of the optimal result in 100-fold experiment.

Function	Standard AFSA			Improved AFSA		
	Max	Min	Mean	Max	Min	Mean
Rastrigin	128	45	81	22	8	16
Easom	27	7	19	11	5	5
Booth	24	5	17	15	4	9
Bohachevsky	30	8	21	13	2	7
Sum Squares	49	17	34	19	6	13



FIGURE 5: Pioneer 3-DX mobile robot.

The Stanford Artificial Intelligence Laboratory (SAIL) with the support of the Stanford AI Robot project firstly developed the ROS (robot operation system) in 2007. It is a Linux

based software framework that is widely used in robotics. This framework uses the idea of nodes, messages, packages, services, and topic [20].

5.2. Experiment Process

(1) *The Self-Created Environment.* The experiment of path planning in autonomous navigation is started in a real environment and a self-created environment, respectively. The self-created environment is shown in Figure 6. This environment consists of some other complicated structures like narrow passes, edges, sharp corners, and so forth. The map of the self-created environment is shown in Figure 7. And points A, B, C, D, and E in the map are the five positions of self-created environment. During the process of path planning, the robot's walk path is represented as its footprint in the map as shown by the blue arrow in Figure 7.

The performance of path planning with the IAFSA application in robot navigation is shown in Figure 8(a). In order to further verify the superiority and effectiveness of IAFSA in robot path planning, IAFSA is compared experimentally with the standard AFSA, A^* algorithm, and HAAFSA, respectively. Figures 8(b), 8(c), and 8(d) correspond to the performance of standard AFSA, A^* algorithm, and HAAFSA. By comparison experiment shown in Figure 8, it can be observed from the results that all the abovementioned algorithms can finish the robot path planning and enable the robot to avoid the obstacles. The optimal performance of IAFSA is better than of the other three algorithms; the global search ability is better and self-adaptive ability is very strong.



FIGURE 6: The self-created environment.

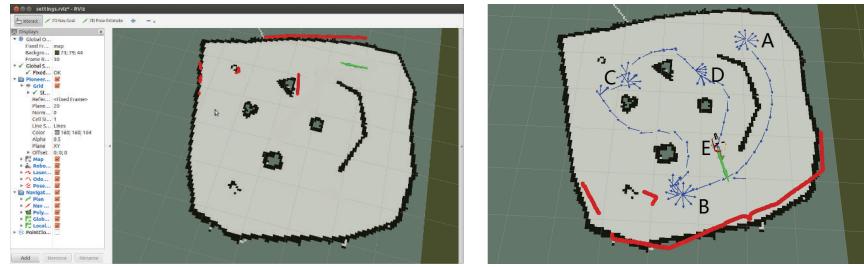


FIGURE 7: The map of self-created environment.

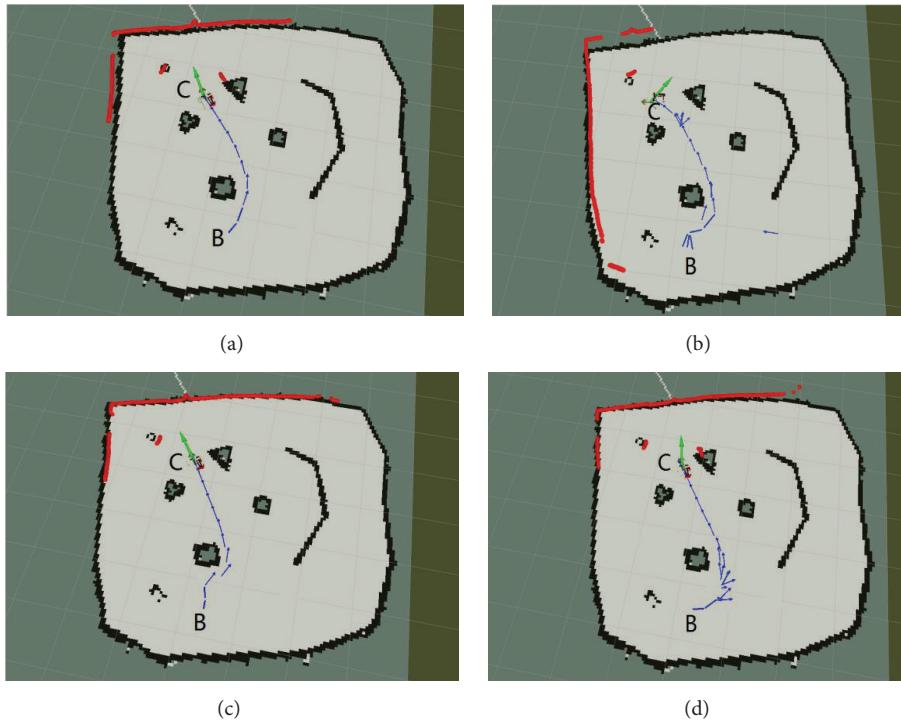


FIGURE 8: The path in self-created environment using different algorithms.

The experiment of path planning is performed 50 times in four different paths, respectively. We get the route data of robot walking in a path from the odometer in Pioneer 3-DX mobile robot, and the average values of route in 50 times are given in Table 4. The odometer in Pioneer 3-DX is used for

speed measurement and route measurement. And, in ROS, the odometer is used for assistant position.

In Table 4, the route obtained by IAFSA is the shortest. The performance of IAFSA in robot path planning is better than of the other three algorithms, and the accuracy of the



FIGURE 9: The real environment.

TABLE 4: The route of the robot path planning experiment in a self-created environment (m).

	A*	AFSA	HAAFSA	IAFSA
A-B	6.658	7.201	6.352	5.624
B-C	4.713	5.593	4.731	3.956
C-D	3.967	4.002	3.989	3.253
D-E	3.594	3.702	3.536	2.821

TABLE 5: The route of the robot path planning experiment in a real environment (m).

	A*	AFSA	HAAFSA	IAFSA
A-B	5.824	6.855	5.875	5.31
A-C	12.621	14.255	12.751	10.982
E-D	3.875	4.254	4.045	3.125
D-B	13.352	15.245	14.145	11.145

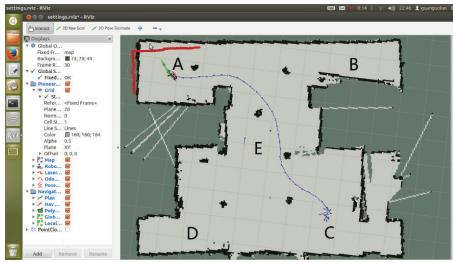


FIGURE 10: The map of the real environment.

result is enhanced. Also, the global search ability and local search ability are improved.

(2) *The Real Environment.* The real environment, which consists of a hall, a cubby, hallways, a cupboard, a booth, and a wall, is shown in Figure 9. And the map of the real environment is shown in Figure 10.

In the map, there are five positions, A, B, C, D, and E, and we choose four paths to do the experiment. The path of the robot is shown in Table 5.

To demonstrate the success of the proposed method, we make another three extra comparisons. The proposed algorithm is compared with the standard AFSA, A* algorithm, and HAAFSA, respectively. Figures 11(a), 11(b), 11(c), and 11(d) correspond to A* algorithm, standard AFSA, HAAFSA,

and IAFSA. Experimental results of path A-C are shown in Figure 11. By comparison, we can find that the four algorithms get better performance in path planning; the mobile robot can pass through the sufficient area and avoid collision with the obstacle. However, the proposed IAFSA in this paper makes the step smaller, the trajectory smoother, and the total route shorter. The performance of IAFSA is better than of the other three algorithms, and the optimized ability is more powerful.

The four algorithms are executed 50 times, respectively, and the mean of the four routes (A-B, A-C, D-B, and E-D) of the robot path planning using the four optimal algorithms is displayed in Table 5.

In Table 5, we can find out that the path got by IAFSA is the shortest, followed by A* algorithm, HAAFSA, and AFSA, respectively. Compared with the other three optimal algorithms, the performance of IAFSA in mobile robot path planning is better. IAFSA is shown to have efficient searching capability in real environment, and the accuracy of the result is enhanced.

6. Conclusion

In this study, an attenuation function α is introduced to improve the weak point of visual in standard AFSA; also, an adaptive operator is introduced to enhance the adaptive step. Besides, a self-active inertia weight factor is introduced to improve the convergence. The IAFSA is applied to the path planning problem of mobile robot which is equipped with ROS. Five well-known benchmark functions are used to evaluate the performance of IAFSA. Simulation shows that the convergence speed of IAFSA is more efficient; the local search ability is enhanced and the optimal result is more accurate. The performance of IAFSA is better than of standard AFSA. In order to further analyze the superiority and effectiveness of the IAFSA in robot path planning, the IAFSA is compared experimentally with the standard AFSA, A* algorithm, and HAAFSA, both in a self-created environment and in a real environment. Experiment results show that IAFSA can adapt to different environments in the planned trajectories with both satisfactory accuracy and stability. The optimal ability is better and the route, which the robot walked, is the shortest. The space complexity is reduced and the path is smoother.

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 add practical applicability. Second, we will extend to multiobjective optimizer to search

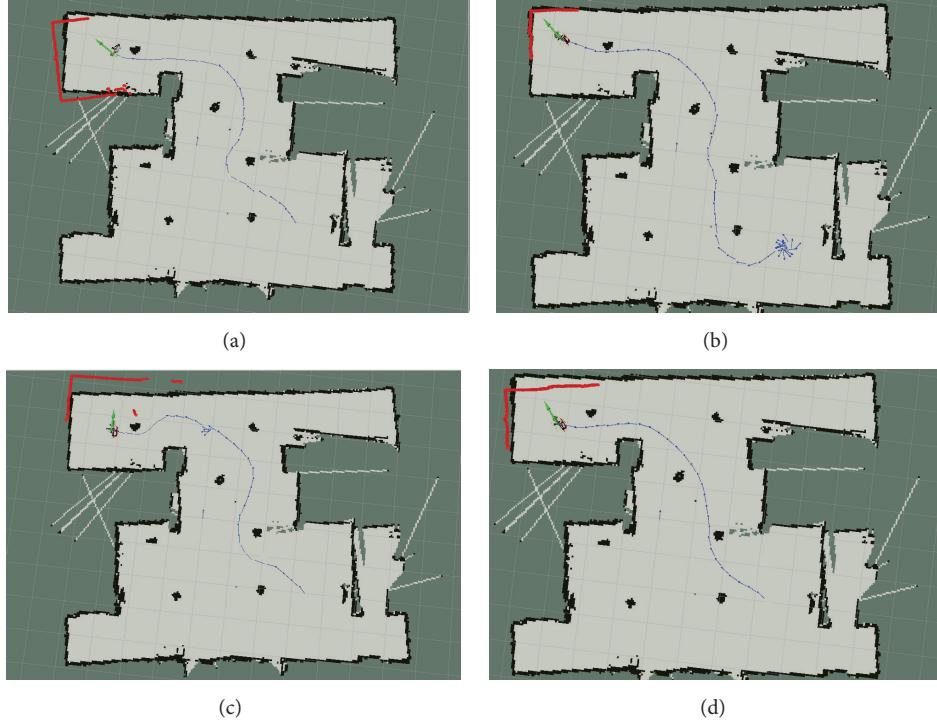


FIGURE 11: The path in the real environment.

the Pareto frontier which can optimize the trade-off between multiple requirements of the robot system. In addition, we are interested in using IAFSA in dynamic environment.

Competing Interests

The authors declare that there are no competing interests regarding the publication of this paper.

Acknowledgments

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

References

- [1] X.-D. Liang, L.-Y. Li, J.-G. Wu, and H.-N. Chen, "Mobile robot path planning based on adaptive bacterial foraging algorithm," *Journal of Central South University*, vol. 20, no. 12, pp. 3391–3400, 2013.
- [2] 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, no. 5, pp. 773–781, 2010.
- [3] A. Tuncer and M. Yildirim, "Dynamic path planning of mobile robots with improved genetic algorithm," *Computers and Electrical Engineering*, vol. 38, no. 6, pp. 1564–1572, 2012.
- [4] Y. Zhang, P. Agarwal, V. Bhatnagar, S. Balochian, and J. Yan, "Swarm intelligence and its applications," *The Scientific World Journal*, vol. 2013, Article ID 528069, 3 pages, 2013.
- [5] 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*, vol. 2014, Article ID 273043, 6 pages, 2014.
- [6] 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, no. 11, pp. 32–38, 2002.
- [7] Q. Ma and X. Lei, "Application of artificial fish school algorithm in UCAV path planning," in *Proceedings of the International Conference on Bio-Inspired Computing: Theories and Applications (Bic-Ta '10)*, vol. 13, no. 1, pp. 93–105, University of Hunan, Changsha, China; Liverpool Hope University, Liverpool, UK, September 2010.
- [8] Z. Ni, X. Zhu, L. Ni, M. Cheng, and Y. Wang, "An improved discrete optimization algorithm based on artificial fish swarm and its application for attribute reduction," *Journal of Information and Computational Science*, vol. 12, no. 6, pp. 2143–2154, 2015.
- [9] T. Yang and T. Yong, "Short life artificial fish swarm algorithm for wireless sensor network," in *Proceedings of the IEEE International Conference on Computational Problem-Solving*, pp. 378–381, Austin, Tex, USA, October 2013.
- [10] R. Azizi, "Empirical study of artificial fish swarm algorithm," *Computer Science*, vol. 17, no. 6, pp. 626–641, 2014.
- [11] 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, no. 6, pp. 2042–2047, 2013.
- [12] 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, no. 8, pp. 1–12, 2014.
- [13] X. Ma, "Application of adaptive hybrid sequences niche artificial fish swarm algorithm in vehicle routing problem," in *Proceedings of the 2nd International Conference on Future Computer and*

Communication (ICFCC '10), pp. V1654–V1658, Wuhan, China, May 2010.

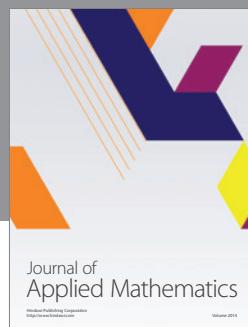
- [14] X. C. Xia and J. Y. Chen, “Three-dimensional sensor node localization based on AFSA-LSS VM,” *International Journal of Control & Automation*, vol. 7, no. 8, pp. 399–406, 2014.
- [15] K. C. Lin, S. Y. Chen, and J. C. Hung, “Feature selection for support vector machines base on modified artificial fish swarm algorithm,” in *Ubiquitous Computing Application and Wireless Sensor*, pp. 297–304, Springer Netherlands, 2015.
- [16] 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, no. 1, pp. 38–53, 2014.
- [17] Y. Gao, L. Guan, and T. Wang, “Triaxial accelerometer error coefficients identification with a novel artificial fish swarm algorithm,” *Journal of Sensors*, vol. 2015, Article ID 509143, 17 pages, 2015.
- [18] Q. D. Qin, L. Li, and R. J. Li, “A novel PSO with piecewise-varied inertia weight,” in *Proceedings of the 2nd IEEE International Conference on Information and Financial Engineering (ICIFE '10)*, pp. 503–506, IEEE, Chongqing, China, September 2010.
- [19] Y.-L. Chen, J. Cheng, C. Lin, X. Wu, Y. Ou, and Y. Xu, “Classification-based learning by particle swarm optimization for wall-following robot navigation,” *Neurocomputing*, vol. 113, no. 7, pp. 27–35, 2013.
- [20] S. Zaman, W. Slany, and G. Steinbauer, “ROS-based mapping, localization and autonomous navigation using a Pioneer 3-DX robot and their relevant issues,” in *Proceedings of the Saudi International Electronics, Communications and Photonics Conference (SIECPC '11)*, pp. 1–5, Riyadh, Saudi Arabia, April 2011.



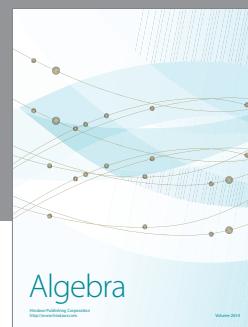
Advances in
Operations Research



Advances in
Decision Sciences



Journal of
Applied Mathematics



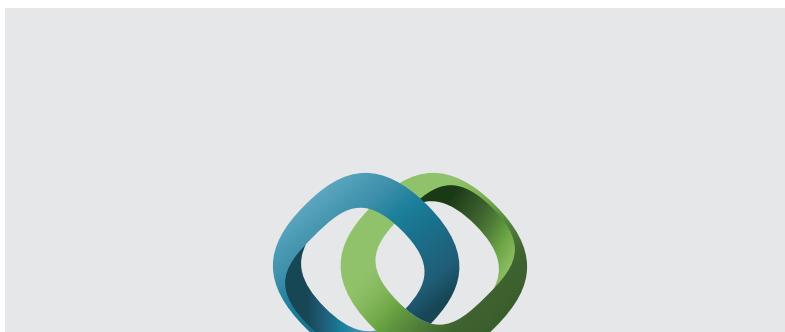
Algebra



Journal of
Probability and Statistics



The Scientific
World Journal

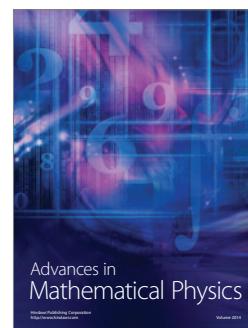


Hindawi

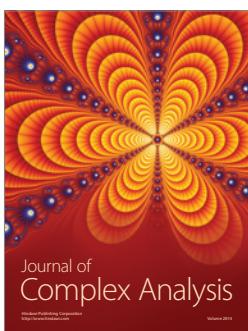
Submit your manuscripts at
<http://www.hindawi.com>



International Journal of
Combinatorics



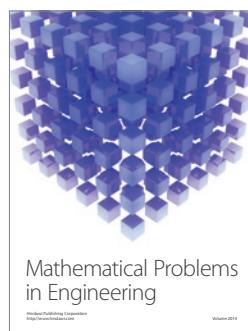
Advances in
Mathematical Physics



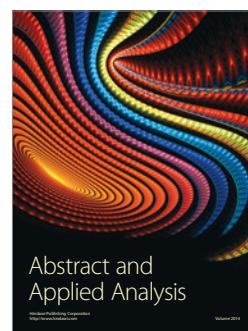
Journal of
Complex Analysis



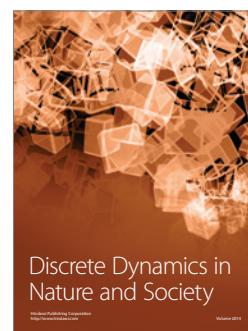
Journal of
Mathematics



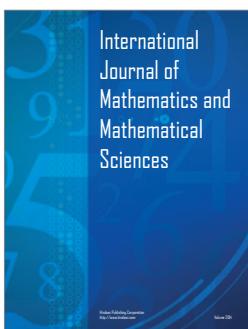
Mathematical Problems
in Engineering



Abstract and
Applied Analysis



Discrete Dynamics in
Nature and Society



International
Journal of
Mathematics and
Mathematical
Sciences



Journal of
Discrete Mathematics



Journal of
Function Spaces



International Journal of
Stochastic Analysis



Journal of
Optimization