Proceeding of the 2015 IEEE
International Conference on Information and Automation
Lijiang, China, August 2015

# A Novel Multi-objective Artificial Bee Colony Algorithm for Multi-robot Path Planning

Zhongya Wang, Min Li*, *Member*, *IEEE*, Lianhang Dou, Yang Li, Qingying Zhao and Jie Li

*School of Mechatronic Engineering and Automation*
*Shanghai University*
*Shanghai, China*
wangzhongya@shu.edu.cn

***Abstract*** **-An improved multi-objective artificial bee colony algorithm is designed in this paper reflecting the multi-objective characteristic of multi-robot path planning problems. Firstly, the foraging mechanism is optimized and a new method to calculate crowding distance is proposed. The restructuring and elimination mechanism of food sources are also presented. Secondly, an improved environment map representation method is presented in which the path information of robots is denoted using Cartesian coordinates directly. Thirdly, three objective functions for path planning are designed according to three key performance indicators which are length, safety and smoothness of the path. Finally, the simulation results show that the improved multi-objective artificial bee colony algorithm can be effectively applied to solve multi-robot path planning problems.**

***Index Terms - Multi-objective artificial bee colony algorithm; multi-robot cooperation; path planning***

## I. INTRODUCTION

With the rapid development of robotics, the performance, robustness and efficiency of individual robots have been improved effectively. However, the need for robot functions is also growing. When we solve some complex, distributed and parallel tasks, the advantages of low cost, high efficiency and robustness of multi-robot system are reflected.

Path planning is one of the basic problems in multi-robot system. Although multi-robot path planning has got considerable development in the past decade, it is still a challenging issue that the path optimization objectives of total length, safety and smoothness are often conflicted.

Researchers from all over the world have proposed a lot of multi-objective optimization algorithms, such as multi-objective genetic algorithm, multi-objective particle swarm optimization and multi-objective immune algorithm. As early as 1985, Schaffer proposed the vector evaluating genetic algorithm(VEGA) in [1], which is regarded as the pioneering work of using evolutionary algorithm to solve multi-objective optimization problems. Afterwards, the multi-objective genetic algorithm(MOGA) in [2], non-dominated sorting genetic algorithm(NSGA) in [3] and niche genetic algorithm (NPGA) in [4] were proposed.

In 2005, an artificial bee colony algorithm was proposed by Karaboga in [5]. The algorithm has been developed greatly in recent years and proved that it has advantages over other heuristic optimization algorithms such as genetic algorithms, particle swarm and differential evolution algorithm [6]. In 2010, Hedayatzadeh *et al.* extended the artificial bee colony algorithm to solve the multi-objective optimization problem in [7]. Omkar *et al.* proposed a multi-objective artificial bee colony algorithm based on vector evaluation aiming at solving optimization problems of the structure combination design in [8,9]. In 2013, Lei proposed a discrete multi-objective artificial bee colony algorithm to solve problems of flexible job-shop scheduling in [10].

In 2014, based on the progressive optimization algorithm, Zhou *et al.* improved the local search strategy of multi-objective artificial bee colony algorithm and solved short-term scheduling problem of electric heating system in [11]. Yahya *et al.* used converged multi-objective artificial bee colony algorithm to solve layout planning problem in [12]. In 2015, Mahapatra *et al.* proposed a discrete artificial bee colony algorithm based on non-dominated sorting and crowding distance selection mechanism to obtain Pareto optimal solution set in [13].

The rest of this paper is organized as follows. Multi−objective artificial bee colony algorithm is introduced in Section II. An improved multi-objective artificial bee colony algorithm is proposed in Section III. Experiment and simulation are discussed in Section IV. The paper is concluded in Section V.

## II. MULTI-OBJECTIVE ARTIFICIAL BEE COLONY ALGORITHM

Artificial bee colony algorithm was designed according to the natural foraging behavior of bees. The foraging model of bees has four constituent elements (food source, three different kinds of bees) and two basic behaviours (recruit honey bees and abandon food sources). The three kinds of bees are employed bees, onlooker bees and scout bees respectively. Food sources represent the potential solutions of the optimization problem. Each food source usually corresponds to an employed bee. Employed bees optimize food sources according to the current information. Onlooker bees search for food sources according to the information provided by

This work is supported by National Nature Science Foundation of China under Grant 61273353.
* Min Li is the corresponding author, e-mail: min_li@shu.edu.cn

978-1-4673-9104-7/15/$31.00 ©2015 IEEE

employed bees in the vicinity of the hive. The food source of the poorest quality is abandoned by scout bees.

Artificial bee colony algorithm relies on self-organizing ability of bees to achieve positive and negative feedback mechanisms [14]. If the nectar quality of a food source is good, the number of corresponding onlooker bees will increase. This is the positive feedback mechanism. If the nectar quality of a food source doesn't improve in a long term, employed bees and onlooker bees will reduce its optimization gradually. Onlooker bees will give up this food source and search for new food sources eventually. This is the negative feedback mechanism.

Multi-objective optimization problem needs to optimize multiple conflicting objective functions. The multi-objective optimization problem which has *m* objective functions and *n* decision variables can be described as:

$$\begin{aligned} &\textit{Minimize} \quad y = (f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \cdots, f_m(\boldsymbol{x})) \\ &\textit{subject to} \begin{cases} h_i(\boldsymbol{x}) \le 0, \ i = 1,2,\cdots, I \\ l_j(\boldsymbol{x}) = 0, \ j = 1,2,\cdots, J \\ \quad \boldsymbol{x} \in D \end{cases} \end{aligned} \tag{1}$$

In the equation, $\boldsymbol{x} \in \boldsymbol{D}$ is *n*-dimensional decision variable, $\boldsymbol{x} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_n)$, $\boldsymbol{D} \subset \boldsymbol{R}^n$ is *n*-dimensional decision space, $\boldsymbol{x}_\mathrm{i} \in \boldsymbol{x}$ represents the $i$ decision variable; $h_i(\boldsymbol{x}) \le 0$ is an inequality constraint, $l_j(\boldsymbol{x}) = 0$ is an equality constraint.

Usually, it doesn't exist a single feasible solution for multi-objective optimization problem, but a set of feasible solutions. There are mainly two ways to solve multi-objective optimization problems. One is to transform the multi-objective optimization problem into a single objective optimization problem by weighting method. The other is to obtain a set of solutions which aren't dominated by each other through Pareto dominance relationship. Most modern multi-objective decision making theories use Pareto dominance relationship to get the solution set of multi-objective optimization problems. The concept of Pareto dominance is:

$\forall\, i = 1,2,\cdots, m,\ f_i(x_a) \le f_i(x_b)$, and $\exists\, j \in \{1,2,\cdots, m\}$, $f_i(x_a) < f_i(x_b)$, then compared $x_a$ and $x_b$, $x_a$ is Pareto dominant. Denoted that $x_a \prec x_b$ means $x_a$ dominants $x_b$.

Multi-objective artificial bee colony algorithm includes five basic steps, which are generation of food sources, food exploration by employed bees, food exploration by onlooker bee, food source evaluation mechanism and food exploration by scout bees. Wherein, the best food sources are stored by an external archive.

*1) Generation of food sources.* The first step is parameter initialization in the generation of the food sources. A set of random solutions $(\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_i, \cdots, \boldsymbol{x}_N)$ are generated and the external archive is updated. Vector $\boldsymbol{x}_i$ is the solution to be optimized during every iteration process. The initialization method is:

$$x_{ij} = rand(0,1) \times (ub_i - lb_i) + lb_i \tag{2}$$

In the equation, $ub_i$ is the maximum of $x_{ij}$, $lb_i$ is the minimum of $x_{ij}$.

*2) Food exploration by employed bees.* Based on the current food source information $\boldsymbol{x}_i$, employed bees search for new food source $\boldsymbol{x}_i'$. The generation method of individual food source $\boldsymbol{x}_i'$ is as follows:

$$x_{ij}' = \begin{cases} ub_i & x_{new} > ub_i \\ lb_i & x_{new} < lb_i \\ x_{new} & lb_i < x_{new} < ub_i \end{cases} \tag{3}$$

In the equation, $x_{new} = x_{ni} + \varepsilon_{ni}(x_{ni} - x_{ki})$ is the newly generated food source.

After the generation of food sources, the judging method is applied. If $\boldsymbol{x}_i \prec \boldsymbol{x}_i'$, then enter the next step; otherwise, the external archive is updated. If $\boldsymbol{x}_i$ and $\boldsymbol{x}_i'$ aren't dominated by each other, the external archive is also updated.

*3) Food exploration by onlooker bees.* Onlooker bees search for new food sources based on current information $\boldsymbol{x}_i$ and their own preferences. Not all food sources are updated in the process of onlooker bees exploration. The probability of selecting every food source by onlooker bees can be calculated as:

$$p_i = \frac{f_t(\boldsymbol{x}_i)}{\sum_{i=1}^{N} f_t(\boldsymbol{x}_i)} \tag{4}$$

If the food source is in line with the onlooker bees' expectation, onlooker bees will choose this food source to explore. After this step, onlooker bees will also update the external archive according to Pareto domination relationship.

*4) Food exploration by Scout bees.* Scout bees judge current food source information to determine which food source to be abandoned and generate a new food source randomly.

$$\boldsymbol{x}_i^{k+1} = \begin{cases} \boldsymbol{x}_{new} & trial > limit \\ \boldsymbol{x}_i^k & trial \le limit \end{cases} \tag{5}$$

In the equation, *trial* is the foraging number, *limit* is the largest selection times, $\boldsymbol{x}_{new}$ represents the newly generated food source.

*5) Food source evaluation mechanism.* Food source evaluation mechanism is one of the most important steps of multi-objective artificial bee colony algorithm and also an important factor to affect the performance of the algorithm. Evaluation mechanism commonly uses Pareto dominance relationship and the crowding distance of food sources.

## III. Improved Multi-Objective Artificial Bee Colony Algorithm

This paper optimizes the multi-objective artificial bee colony algorithm to improve its performance aiming at multi-robot path planning task. The improvements are as follows.

*1) Optimization of foraging mechanism*

Every employed bee would generate a new solution in the iteration process of multi-objective artificial bee colony algorithm. The newly generated solution is compared with the original solution to decide which one to be retained. Similarly, the foraging process of every onlooker bee executes the comparison of solutions. The solutions that both of bees accept will affect the efficiency of multi-objective artificial bee colony algorithm greatly. In this case, the algorithm is not efficient.

The improved multi-objective artificial bee colony algorithm optimizes the foraging mechanism of bees. In each iteration process of the algorithm, the bees generate a new food source through exploration process firstly. Then the newly generated food sources are mixed with the previous food sources. Finally, the recombined food sources are sorted based on Pareto dominance and crowding distance. According to the sorted result, the food sources of poor quality are abandoned to maintain a constant number of food sources. As a result, the algorithm only needs one time of the sorting process in each iteration process which improves the operational efficiency of the algorithm.

Scout bees have two roles in the algorithm. One is to abandon the food sources of poor quality to accelerate the convergence rate. The other is to produce new food sources to increase the diversity of the solutions. The foraging process of scout bees is put before that of employed bees and onlooker bees in the improved multi-objective artificial bee colony algorithm. This improvement has two advantages. On one hand, the algorithm modifies the selection strategies of employed bees and onlooker bees. These two kinds of bees only explore new food sources without updating them, which greatly increases the importance of the referenced food sources which are provided by the scout bees. On the other hand, scout bees can abandon the food source of poor quality and generate a new food source. It is efficient for employed bees and onlooker bees to optimize this food source.

The improved foraging mechanism is shown in algorithm 1.

*2) Calculation of crowding distance*

The calculated maximum, minimum and average value of each objective function will vary greatly for different optimization objectives in multi-objective optimization problem. If we don't take this situation into account when calculating crowding distance, different objectives may have different optimization degree. To avoid this situation, the calculated crowding distances are subjected to the normalization process in the improved multi-objective artificial bee colony algorithm. Normalization process is shown in equation (6).

$$d_i' = \frac{d_i}{d_{dist}^{Max} - d_{dist}^{Min}} \tag{6}$$

Algorithm 1： The improved foraging mechanism

Begin
1. for $i = 1$ to *maxCycle*
2. $\boldsymbol{x}_s' = scout(\boldsymbol{x}_i)$ //onlooker bees exploration
3. $\boldsymbol{x}_e' = employed(\boldsymbol{x}_s')$ // employed bees exploration
4. $\boldsymbol{x}_o' = onlooker(\boldsymbol{x}_s')$ //onlooker bees exploration
5. $\boldsymbol{x}_i' = \boldsymbol{x}_e' + \boldsymbol{x}_o' + \boldsymbol{x}_s'$ // recombination of food sources
6. $\boldsymbol{x}_i'' = nonDominationSort(\boldsymbol{x}_i')$ //sorting
7. $\boldsymbol{x}_i = sort(\boldsymbol{x}_i'')$ //retain good food sources
8. endfor

End

In the equation, $d_i'$ is the normalized crowding distance, $d_i$ is the crowding distance before normalization, $d_{dist}^{Max}$ and $d_{dist}^{Min}$ are the maximum and minimum crowding distance values after eliminating the crowding distance of infinite value.

The scope of the crowding distance values of all objective functions is located between [0,1] (except for the infinite case) after normalization. The optimization criteria of all objectives are unified.

*3) Restructuring and elimination mechanism of food sources*

The improved multi-objective artificial bee colony algorithm doesn't use an external archive to preserve good food sources. The restructuring and elimination mechanism of food sources are designed in this paper to ensure excellent ones to be reserved. The restructuring process of the food sources is to regroup the new food sources which have been generated by scout bees, onlooker bees and employed bees. The food source elimination mechanism is carried out based on non-dominated sorting and crowding distance calculation results. Firstly, the food sources whose dominance level is 1 are reserved. The food sources are reserved according to crowding distance if the number exceeds the initial food sources. Then the food sources of dominance level 2 are added until the initial number of food sources is reached. Other food sources are discarded.

*4) Environment map building*

The commonly used approach to environment map modeling is grid method. This paper uses the Cartesian coordinate system to establish environment map in order to avoid the complexity of the grid division. The motion path of the robot $(p^x, p^y)$ can be determined by the angle $\theta_i$ between the start point $(S^x, S^y)$ of the robot and the robot path point $(p_k^x, p_k^y)$ assumed that the robot step value is a constant.

The optimization objective is transformed from path point $(p_k^x, p_k^y)$ to angle $\theta_i$ through equation (7).

$$\begin{cases} p_k^x = S^x + \sum_{i=1}^{k} l \times cos(\theta_i) \\ p_k^y = S^y + \sum_{i=1}^{k} l \times sin(\theta_i) \end{cases} \tag{7}$$

*5) Optimization of path evaluation mechanism*

We propose three objective functions according to the three path performance indicators of path length, safety and smoothness. The objective functions are path length function $f_l$, path safety function $f_{sa}$ and path smoothness function $f_{sm}$. Then the multi-robot path planning problem is transformed to solve the objective function in equation (8).

$$\begin{aligned} &F(\boldsymbol{Step}^*) = min\left[ f_l(\boldsymbol{\theta}), f_{sa}(\boldsymbol{\theta}), f_{sm}(\boldsymbol{\theta}) \right] \\ &subject\ to: Step_1^* = S, step_n^* = G, \boldsymbol{Step}^* \cap \boldsymbol{Obs} = \varnothing \end{aligned} \tag{8}$$

$\boldsymbol{Step}^*$ = $\{Step_1^*$, $Step_2^*$, …, $Step_m^*\}$ is the feasible path set; $\boldsymbol{\theta}$ is the transformed path set; $S$ is the starting point of the robot, $G$ is the target point of the robot; $\boldsymbol{Obs}$ is the set of obstacles.

Path length function lets the robots walk from the starting point to the target point according to the shortest path. If the distance between a robot and an obstacle or other robots is too close, the path safety function begins to optimize the path. This ensures that the robots and the obstacles maintain a reasonable safety distance. The definition of the path safety function is shown in equation (9).

$$f_s = \begin{cases} f_{Obs} & f_{Obs} \le L \\ const & f_{Obs} > L \end{cases} \tag{9}$$

In the equation, $L$ is the specified safety distance between robots and obstacles; $f_{Obs}$ is the distance between the robots paths and the obstacles; $const$ stands for the constant distance.

The path smoothness function is defined as equation (10).

$$f_{sm}(\boldsymbol{Step}^*) = max\left\{ \sum_{i=2}^{n} (\theta_i - \theta_{i-1}) \right\} \tag{10}$$

In the equation, $\boldsymbol{\theta}$ is the transformed path set; $n$ is the number of path points.

## IV. Experiment and Simulation

The traditional path planning methods can only generate a feasible path when running once. The improved multi-objective artificial bee colony algorithm for solving multi-robot path planning problem in global environment of this paper can generate multiple feasible paths for robots to choose when running once.

There are a variety of criteria for performance evaluation of multi-objective optimization algorithm. The error rate (ER) and the breadth (E) are adopted in this paper. The error rate measures the probability whether the obtained non-domination solution is the actual Pareto frontier or not. The calculation method is shown in equation(11). If the obtained solution is an actual Pareto frontier elements, then $x_i = 0$. Otherwise, $x_i = 1$.

$$ER = \frac{\sum_{i=1}^{n'} x_i}{n'} \tag{11}$$

The breadth measures non-dominated solutions' distribution in the target space. It is calculated as equation(12). In the equation, $m$ is the number of objectives, $\max(f_i)$ and $\min(f_i)$ are the maximum and minimum values of the obtained non-dominated solution in the objective function $f_i$.

$$E = \sqrt{\sum_{k=1}^{m} \left| \max(f_i) - \min(f_i) \right|} \tag{12}$$

Simulation experiments based on Matlab are carried out to verify the effectiveness of the improved multi-objective artificial bee colony algorithm in multi-robot path planning. The environment map of the simulation is set to $100*100$ pixels. Two experiments have been conducted based on three objective functions which are the path length, safety and smoothness. Experiment I is to verify the effectiveness of the paths generated by one single robot based on the improved multi-objective artificial bee colony algorithm and the performance is analysed by comparison with other methods. In experiment II, there are multiple robots and obstacles.

*A. Experiment I*

A single robot path planning in the environment with one concave obstacle is carried out in order to verify the effectiveness of the improved multi-objective artificial bee colony algorithm. Initialization parameters of the algorithm are as follows. The population size is 100. The maximum number of iterations is 500. The coordinate of the starting point is $(0,0)$ and the coordinate of the target point is (100, 100) .The step length of the robots is 7. The position and shape of the obstacle in the environment are shown in Figure 1. The shape of the obstacle is to set concave in order to verify the effectiveness of the paths in complex environments. The path planning result is shown in Figure 1. As shown in the figure, the robot can well avoid the concave obstacle and find the shortest path from the starting point to the target point.

The path planning of a single robot in the environment with multiple obstacles is carried out to verify the effectiveness of the improved algorithm. Initialization parameters of the algorithm are as follows. The population size is 100. The maximum number of iterations is 2000. The coordinate of the starting point is $(0,0)$ and the coordinate of the target point is $(100, 100)$. The step length of the robots is 7. There are three obstacles in the environment which are indicated by red dots. Its location coordinates are $(30,40)$, $(50,50)$, $(70,60)$. The path planning result is shown in Figure 2.

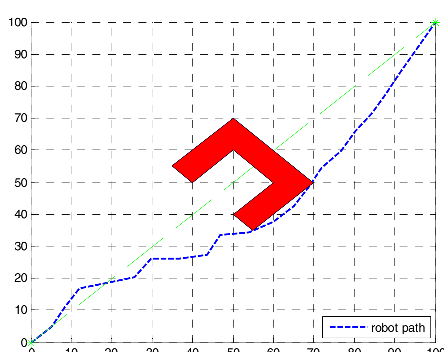


Fig. 1 Path planning of single robot based on multi-objective artificial bee colony algorithm

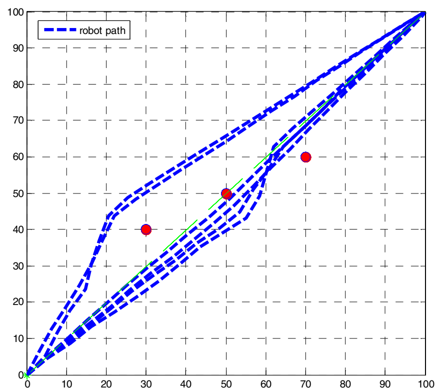


Fig.2 Path planning result

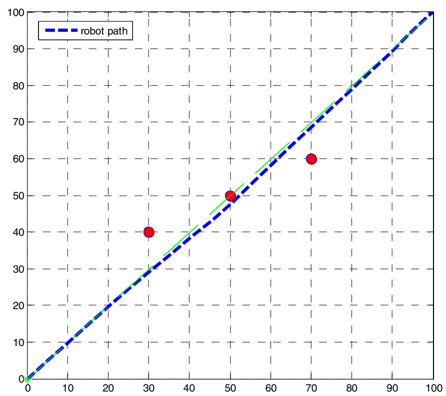


Fig.3 The shortest path

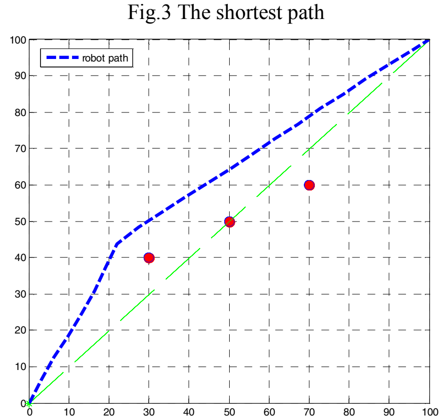


Fig.4 The safest path

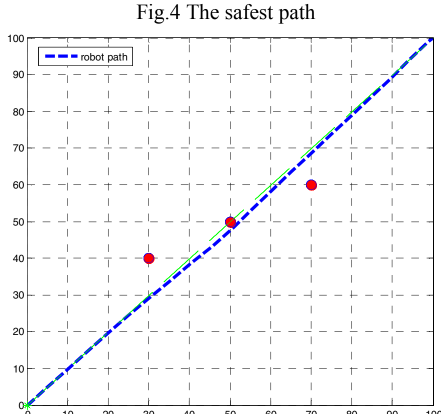


Fig.5 The smoothest path

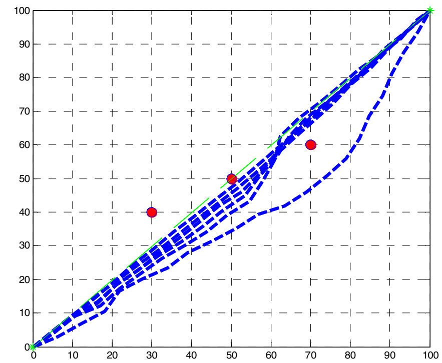

Fig.6 Path planning results of NSGA-II

When the algorithm runs once, multiple feasible paths are generated considering as the path length, safety and smoothness. Figures 3, 4, 5 are the shortest paths, the safest paths and the smoothest paths. The robot can choose a path based on the current decision-making criterion.

The path planning experiment using NSGA-II algorithm in the same environment are carried out to compare the performance of the improved multi-objective artificial bee colony algorithm with other multi-objective optimization algorithms. The parameter setting of NSGA-II algorithm is as follows. The population size is 50. The number of cycles is 2000. The crossover rate is 0.9 and the mutation rate is 0.1. Figure 6 is the path planning result using NSGA-II algorithm. By comparing Figur2 with Figure 6, the path diversity and smoothness of the proposed algorithm are better than that of NSGA-II. For the quantitative analysis of the two path planning results, each algorithm has been executed for 10 times. Table I shows the mean value of the error rate and breadth of the path planning results.

It can be observed from Table I that the multi-objective artificial bee colony algorithm performs better than NSGA-II algorithm in breadth and error rate.

### B. *Experiment II*

There are multiple robots and obstacles in this path planning experiment. The environment map of the simulation is set to 100*100 pixels. The step length of the robots is 7. Initialization parameters of the algorithm are as follows. The population size is 100. The maximum number of iterations is 2000. Three robots and four obstacles are deployed in the experiment. The coordinate of the starting points and the target points are shown in Table II.

TABLE I

Error rate and breadth comparison of the algorithms

| Evaluation Indicator | Multi-objective Artificial Bee Colony Algorithm | NSGA-II |
|---|---|---|
| Breadth | 37.6823 | 36.9044 |
| Error rate | 0 | 0.2% |

TABLE II

Start and destination coordinates of the robots

| Robot Number | Starting Point Coordinates | Target Point Coordinates |
|---|---|---|
| 1 | (10,10) | (90,90) |
| 2 | (20,40) | (85,55) |
| 3 | (15,90) | (80,20) |

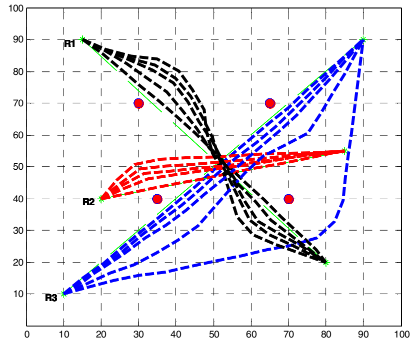

Fig.7 Path planning results of multi-robot

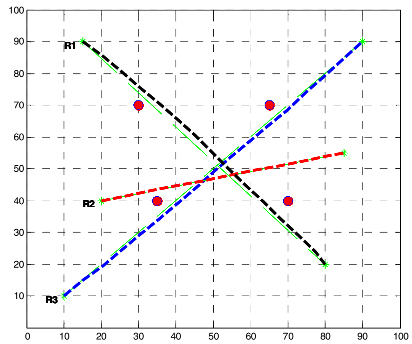

Fig.8 The shortest path

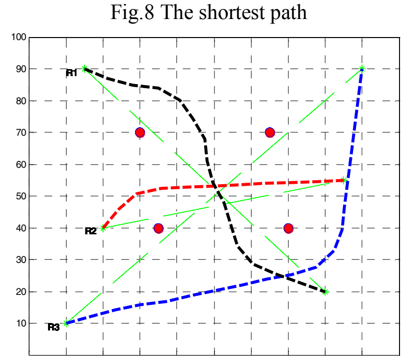

Fig.9 The safest path

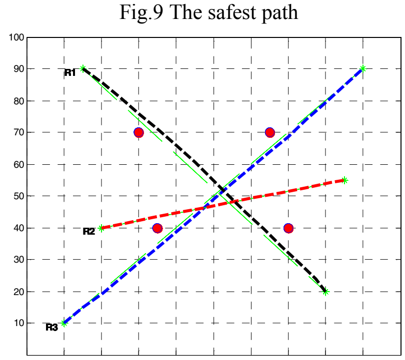

Fig.10 The smoothest path

The coordinates of the obstacles are $(35,40)$, $(30,70)$, $(65,70)$ and $(70,40)$. The path planning results are shown in Figure 7. Multiple paths are generated for the three robots to choose after the algorithm runs once. The figure shows a part of the result of the planned paths. The number of the generated paths depends on the size of the initial population. Figures 8, 9 and 10 are the shortest, the safest and the smoothest paths of the results respectively.

The path planning results above show that the improved multi-objective artificial bee colony algorithm can obtain a number of optimization paths corresponding to different indictors at the same time. The robots can select the desired path based on the current mission requirements and their own preferences. There are many different combinations of paths according to the selection of the robots.

## V. Conclusion

This paper presents an improved multi-objective artificial bee colony algorithm for multi-robot path planning problem. In order to get a better performance, the improvements are made according to the multi-objective characteristic of multi-robot path planning problems. Simulation results show that the improved multi-objective artificial bee colony algorithm can effectively complete the multi-robot path planning task with a convincing performance.

## References


[1] Schaffer J D. Multiple objective optimization with vector evaluated genetic algorithms[C] Proceedings of the 1st international Conference on Genetic Algorithms. L. Erlbaum Associates Inc., 1985: 93-100.

[2] Fonseca C M, Fleming P J. Genetic Algorithms for Multiobjective Optimization: Formulation Discussion and Generalization[C] ICGA. 1993, 93: 416-423.

[3] Srinivas N, Deb K. Muiltiobjective optimization using nondominated sorting in genetic algorithms[J]. Evolutionary computation, 1994, 2(3): 221-248.

[4] Horn J, Nafpliotis N, Goldberg D E. A niched Pareto genetic algorithm for multiobjective optimization[C] Proceedings of the First IEEE Conference on Evolutionary Computation,1994: 82-87.

[5] Karaboga D. An idea based on honey bee swarm for numerical optimization[R]. Technical Report, Computer Engineering Department, Engineering Faculty, Erciyes University, 2005.

[6] Karaboga D, Basturk B. On the performance of artificial bee colony (ABC) algorithm [J]. Applied soft computing, 2008, 8(1): 687-697.

[7] Hedayatzadeh R, Hasanizadeh B, Akbari R, et al. A multi-objective artificial bee colony for optimizing multi-objective problems[C] 2010 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE), 2010: 277- 281.

[8] Omkar S N, Senthilnath J, Khandelwal R, et al. Artificial Bee Colony (ABC) for multi-objective design optimization of composite structures [J]. Applied Soft Computing, 2011, 11(1): 489-499.

[9] Omkar S N, Naik G N, Patil K, et al. Vector evaluated and objective switching approaches of artificial bee colony algorithm (abc) for multi-objective design optimization of composite plate structures [J]. International Journal of Applied Metaheuristic Computing (IJAMC), 2011, 2(3): 1-26.

[10]Lei D. Multi-objective artificial bee colony for interval job shop scheduling with flexible maintenance[J]. The International Journal of Advanced Manufacturing Technology, 2013, 66(9-12): 1835-1843.

[11]Zhou J, Liao X, Ouyang S, et al. Multi-objective artificial bee colony algorithm for short-term scheduling of hydrothermal system[J]. International Journal of Electrical Power & Energy Systems, 2014, 55: 542-553.

[12]Yahya M, Saka M P. Construction site layout planning using multi-objective artificial bee colony algorithm with Levy flights[J]. Automation in construction, 2014, 38: 14-29.

[13]Mahapatra K, Nayak M R, Rout P K. Multi-objective Discrete Artificial Bee Colony Based Phasor Measurement Unit Placement for Complete and Incomplete Observability Analysis[M]. Intelligent Computing, Communication and Devices. Springer India, 2015: 255-266.

[14]Abu-Mouti F.S, El-Hawary M.E. Overview of Artificial Bee Colony (ABC) algorithm and its applications [C] IEEE Transactions on Power Systems, 2012: 1-6.