

Mobile Robot Path Planning Using Ant Colony Algorithm

Yuanliang Zhang ^{1*, a, b}, Cheng Chen ^{2, b} and Qing Liu ^{3, b}

^a School of Automation, Southeast University

^b Jiangsu Institute of Marine Resources, Huaihai Institute of Technology

^{1*} Corresponding Author: zhangyuanl@hhit.edu.cn, ² 602748579@qq.com, ³ newlion79@gmail.com

Abstract

Path planning for the mobile robot is to find a shortest obstacle free path from the starting position to the target position. Ant colony optimization method is frequently used to obtain the optimal path in the static known environment. But local minimum and slow convergence are the main problems of ant colony algorithm. This paper proposes a modified ant colony algorithm for path planning of the mobile robot in a known static environment. The modified ant colony algorithm can enlarge the searching range so that the local minimum problem can be weakened, while the algorithm can also converge quickly. And in the optimal path searching process, the turning factor is considered, too. The obtained optimal path has not only short distance, but also few big turning positions. Simulations are done to verify the proposed modified ant colony algorithm.

Keywords: *ant colony algorithm; mobile robot; path planning; static known environment*

1. Introduction

Nowadays, mobile robots have become an important part of production process automation. Mobile robots can complete the corresponding work instead of humans in the dangerous areas where humans cannot enter or work safely because of the geographical conditions, environmental conditions, or other limiting factors. For these reasons, the research on mobile robots obtains extensive attention in the science and engineering areas. Generally speaking, mobile robots work in a known environment or perceive their working environment using sensors. There also exist obstacles in the working places. First of all, the mobile robots need to find a short and obstacle free path from the starting point to the target point. Sometimes, the mobile robots have some other requirements, such as fewer turnings, distance from the obstacles, *etc.*

The precondition for the mobile robots to work safely and smoothly is to get an optimal path in the working environment. The task of the path planning algorithm is to obtain an optimal path to the given goal. As long as the development of the mobile robot technology, path planning for mobile robots obtains extensive research [1-3]. Rashid *et al.* introduced a visibility binary tree algorithm for mobile robots path planning [4]. This algorithm relied on the construction of the set of all complete paths between the robot and target taking into account inner and outer visible tangents between the robot and the circular obstacles. Duchon *et al.* modified the well-known A star algorithm for path planning of a mobile robot based on a grid map [5]. The modified A star algorithm could get the optimal path using lesser computation time. Azimirad and Shorakaei provided a two-stage analytical-evolutionary algorithm considering dynamic equations to find the optimal path for mobile robots [6]. The analytical method was based on the indirect open loop optimal control problem and the evolutionary method was based on genetic algorithm. Dakulović and Petrović dealt with the path planning problem using the

two-way D* algorithm [7]. This algorithm was inspired by the Witkowski's algorithm and based on a two-dimensional occupancy grid map of the environment. Hawa proposed a modified A star algorithm for path planning of mobile robots [8]. This modified algorithm could expand fewer nodes than A star during the search process, especially in scenarios where there are complex-shaped obstacles in the path between the start and goal nodes.

At present, the commonly used path planning methods are artificial potential field method [9], fuzzy logic method [10], A star algorithm [8], and neural network method [11]. The artificial potential field method usually can provide a safe and smooth path. But it is easy to fall into the local minimum solution. To a certain extent fuzzy logic method has some intelligence. The disadvantage of fuzzy logic method is the weak adaptability. It can only provide good solutions for a particular kind of problem. A star algorithm can provide good path planning results, but it is not efficient when the environment is big. Neural network has strong learning ability and adaptability by offline and online training. But how to design the structure is a difficult job that must be done. Obtaining a well-trained neural network is not easy, either.

Some researchers applied the swarm intelligence thought to the path planning task. Espelosín *et al.* dealt with the path planning problem of a mobile robot in two-dimensional sceneries using a flock of moving particles behavior based method [12]. Marco *et al.* proposed an approach combining the artificial bee colony algorithm and the evolutionary programming algorithm to solve the mobile robot path planning problem [13]. Ant colony algorithm is frequently used to solve various kinds of optimization problems, including mobile robots path planning [14-16]. But ant colony algorithm has the local minimum and slow convergence problems. Many studies have been presented to solve these problems. But most of them only focus on one of these two issues. This paper proposes a modified ant colony algorithm to solve the local minimum and slow convergence problems for mobile robots path planning in a static environment. By adding the turning angle operator in calculating the heuristic factor the modified ant colony algorithm can provide the optimal path with short length and fewer turnings.

The following paper is organized as follows: Section 2 presents the ant colony algorithm for mobile robots path planning; Section 3 gives the details about the modification of the ant colony algorithm; Section 4 evaluates the proposed modified ant colony algorithm in an illustrative example and is followed by concluding remarks provided in Section 5.

2. Ant Colony Algorithm for Path Planning

According to optimization problems, there exist many optimization algorithms. Generally, the optimization algorithms can be classified as complete optimization algorithms and approximate optimization algorithms. The complete optimization algorithms can provide optimal solutions for optimization problems with limited size. But the complete optimization algorithms need lots of computation time for complex problems which makes them unsuitable for real applications. Approximate optimization algorithms can save the computation time and obtain extensive attention. Ant colony algorithm is a kind of approximate optimization algorithm which is frequently used in various applications.

Ant colony algorithm is inspired by the foraging behavior of real ant colony. In particular, the ant colony can find the shortest path from the nest to the food source. During the process of searching food, the ants randomly hunt firstly. Then the ants can leave the pheromone in the path they passed. The other ants can detect the pheromone. When the ants choose their paths, they will choose the path with

stronger pheromone at a certain probability. On the way back to their nest, the amount of pheromone left by the ants in the path is different based on the quantity and quality of the food source. The pheromone left in the path can direct other ants to find this food source. The indirect communication among the ant colony basing on the pheromone can make the ant colony find the shortest path from the nest to the food source. In the ant colony model the difference between the artificial ant colony and the real ant colony can be described as follows. (1) The real ants are asynchronous in the environment, while the artificial ants are synchronous. That means that in each searching circulation the path the artificial ants pass from the food source to their nest is the same as that from the nest to the food source. (2) The real ants will leave the pheromone in the path they passed, while the artificial ants only leave the artificial pheromone in the path from the food source to their nest.

Ant colony algorithm was used to solve the travelling salesman problem (TSP) firstly. In this application the task of each ant is to complete a solution of TSP, *i.e.* realize a tour. According to optimization problem, a finite solution set for this optimization problem must be obtained firstly. Then define the pheromone values set, which is usually called pheromone model. From a technical point of view, this set is a kind of parameterized probability model. It is one of the core components of heuristic ant colony algorithm. In general, ant colony algorithm does the following two iterative steps to deal with optimization problems. (1) Build the solution candidate set using the pheromone model, *i.e.* build the parameterized probability distribution in the whole solution domain. (2) Adjust the artificial ant pheromone values using the whole solution set. The target of updating the pheromone values is to compress the searching space in a high quality solutions set. In particular, intensifying the composition of the solution basing on the solution quality is the important factor of ant colony algorithm. That means that good solution has good solution composition.

Ant colony algorithm is a kind of iteration optimization algorithm. Each iteration process includes three steps: solution establishment, pheromone update and background operation. During the solution establishment process, the artificial ants have the heuristic feature. Pheromone update rule is an important part of ant colony algorithm. It consists of two parts. The first one is the pheromone volatilization process in which all of the pheromone values decrease. This process can avoid ant colony algorithm from converging too fast so that the global optimal solution cannot be obtained. The second one is the pheromone increase process. Many pheromone increase rules exist based on different modification versions of ant colony algorithm. Generally speaking better solution has the priority to get more pheromone increment. The aim of background operation is to deal with all of the solutions after an iteration, which cannot be done by single ant.

Usually, when doing path planning for mobile robots using ant colony algorithm grid map is used [17]. By using grid map the environment can be expressed with a two-dimensional matrix, which can simplify the calculation. In this paper the grid map is used to describe the environment. The number of the artificial ants is w . At the beginning of the searching process the pheromone values of all of the paths are set as the same value, $\tau_{i,j}(0) = C$. Here $\tau_{i,j}(0)$ is the pheromone value of the path between point i and j at the time $t=0$ sec and C is a constant. During the searching process the artificial ant k ($k=1,2,\dots,w$) determines the direction of next step based on the pheromone value of each feasible path. The probability of the artificial ant k to choose the next point j at time t can be calculated using Eq. (1) [18].

$$p_{i,j}^k(t) = \begin{cases} (\tau_{i,j}^\alpha(t)\eta_{i,j}^\beta(t)) / \left[\sum_{\gamma \in S_i^k} (\tau_{i,\gamma}^\alpha(t)\eta_{i,\gamma}^\beta(t)) \right], & j \in S_i^k \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

where $\eta_{i,j}(t)=1/d_{i,j}$ is heuristic function, $d_{i,j}$ is the distance between point i and j , α and β are the parameters which reflect the influence of $\tau_{i,j}(t)$ and $\eta_{i,j}(t)$ to the probability $p_{i,j}^k(t)$, and S_i^k is the feasible points set of the ant k .

Pheromone gradually volatilizes with time. The residue level of pheromone in a path can be expressed using the parameter ρ . Assume that the artificial ant completes a searching iteration using h time intervals. Then the pheromone value on this path can be globally modified using Eq. (2).

$$\tau_{i,j}(t+h) = \rho\tau_{i,j}(t) + \sum_{k=1}^W \Delta\tau_{i,j}^k \quad (2)$$

where $\Delta\tau_{i,j}^k$ is the pheromone value left by the artificial ant k in the path (i,j) and W is the number of the artificial ants passed the path (i,j) . $\Delta\tau_{i,j}^k$ can be calculated using Eq. (3).

$$\Delta\tau_{i,j}^k = \begin{cases} \frac{Q}{L_k}, & \text{if the ant go through the path (i,j) in the kth cycle} \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

Here Q is a constant which denotes the strength of the pheromone, and L_k is the length of the path searched by the artificial ant in this iteration.

The calculation steps of ant colony algorithm can be summarized as follows.

- (1) At the beginning, generate the paths that the artificial ants will go.
- (2) Calculate the length of the paths and then calculate the pheromone increment using Eq. (3). Update the pheromone values of the paths using Eq. (2).
- (3) Modify the searched paths.
- (4) Generate the paths from the starting point to the target point based on the transition probability calculated by using Eq. (1) and roulette method, which is introduced in the following. Then switch to step 2.
- (5) End of the algorithm and output the optimized path.

The roulette method is described here. As shown in Figure 1, assume that the artificial ant is in the point D , points A , B and C are the points that the ant has passed, and points E , F and G are the candidates that the ant can choose. Assume that the probabilities of the ant to choose the points E , F and G are $p^E = 24\%$, $p^F = 60\%$ and $p^G = 16\%$, respectively.

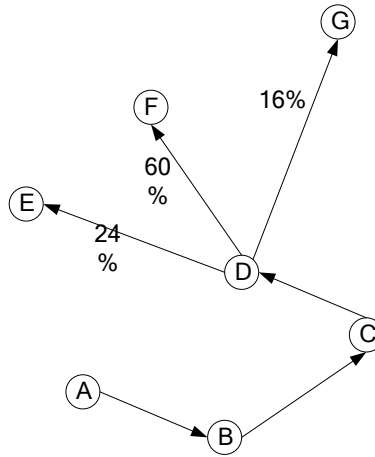


Figure 1. Roulette Method Explanation 1

If the ant chooses point F which has the biggest probability, the algorithm will lose diversity and trap in the local minimum solution easily. By using roulette method this problem can be solved. Select a random number R in the region $[0,1]$. Subtract p^E from R . If the result is less than or equal to 0, the ant will choose the point E . If the result is bigger than 0, subtract p^F from the result continuously. If the result is less than or equal to 0, the ant will choose the point F . Else the ant will choose the point G . As shown in Figure 2, if the arrow turns 0.9 circle point G will be selected.

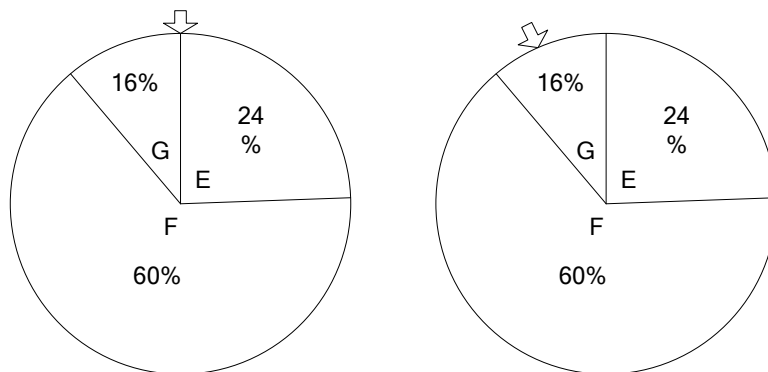


Figure 2. Roulette Method Explanation 2

Figure 2 shows that the probability that the arrow stops in F region is the biggest, and then E and G region. By using roulette method the point with high probability will have bigger opportunity to be selected. At the same time the ant also may select the point with small probability, which makes the algorithm avoid converging too fast.

3. Modified Ant Colony Algorithm

Ant colony algorithm has the following characteristics. (1) Ant colony algorithm has the distributed computation characteristic. It has a strong global searching ability and can provide the global optimal solution in most instances. And the distributed computation characteristic makes ant colony algorithm robust during the searching process. (2) Ant colony algorithm has the self-organization characteristic. This characteristic can increase the robustness of ant colony algorithm. (3) Ant

colony algorithm has positive feedback characteristic. It can enhance the system's stability and reliability, and improve ant colony algorithm's searching efficiency.

Ant colony algorithm has extensive application for many optimization problems. But it also has some disadvantages, including stagnation problem, local minimum, slow convergence and path with many turns. The stagnation problem happens when the artificial ant enter a concave space so that it cannot move next step. The local minimum problem comes from ant colony algorithm's positive feedback characteristic. When a lot of artificial ant converge to some local optimal path the pheromone value in this path will become very big, which can make more artificial ants select this path. In this case less and less artificial ants move to search other paths, so that the algorithm converges to this local optimal solution. Slow convergence and local minimum problem are also two main problems needed to be solved. The two problems are interactional. Usually, in order to avoid trapping in the local optimal solution the searching diversity is improved. This will slow down the convergence velocity. Vice versa, if you want to speed up the algorithm convergence velocity the searching range must be narrowed. Sometimes the optimal path searched using ant colony algorithm has many turns. For heavy loading mobile robots with high speed the path with many turns means much energy consumption and low efficiency. In this case short path with many turns is not a good choice. The mobile robots need a short path with few turns.

In this paper a modified ant colony algorithm is proposed for mobile robots path planning. The modification focuses on the four problems aforementioned. In this modified algorithm when the ant falls into the trap it will go back to the last point and set the trap point as the "not to go" point so that other ants will not fall into this trap again. The modified ant colony algorithm is presented in the following.

(1) Set the number of the artificial ants as m . Assume P_{start} is the starting point, and P_{end} is the end point. At first, all of the artificial ants start from P_{start} . Put P_{start} into the taboo collection $tabu_k$ ($k=1,2,\dots,m$). $tabu_k$ is the points collection passed by the artificial ant k . It is dynamically adjusted during the searching process. The initial pheromone value between point i and j is $\tau_{ij}(0) = \tau_0$ (τ_0 is a constant). Set the value of the iteration counter, NC , and NC_{max} is the maximum iterations. Set the values of the parameters α , β and ρ .

(2) For the artificial ant k , the following rules are used to choose the next point to pass. At time t the probability for the artificial ant k in the point p_i to choose the next point p_j can be calculated using Eq. (4).

$$p_{i,j}^k(t) = \begin{cases} (\tau_{i,j}^\alpha(t)\eta_{i,j}^\beta(t)) / \left[\sum_{\gamma \in \mathcal{S}_i^k} (\tau_{i,\gamma}^\alpha(t)\eta_{i,\gamma}^\beta(t)) \right], & j \in KS_i^k \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

where $\eta_{i,j}(t)$ is local visibility heuristic function and the parameters α and β are the affecting weights of $\tau_{i,j}(t)$ and $\eta_{i,j}(t)$ to $p_{i,j}^k(t)$, respectively. KS_i^k is the feasible points collection that the artificial ant k in the point p_i can choose next step. If $\eta_{i,j}(t) = 1/d_{i,j}$, the heuristic function only consider the distance factor. So that the obtained optimal path may have many turns which is not a good path in some cases. In this paper the distance and turning factors are both considered when calculate $\eta_{i,j}(t)$. $\eta_{i,j}(t)$ is calculated using Eq. (5).

$$\eta_{i,j}(t) = \frac{h_{angle}}{d_{i,j}} \quad (5)$$

where h_{angle} is the turning operator. In the grid map environment there are four kinds of turning angle, 45° , 90° , 135° and 180° . The value of h_{angle} is set based on the turning angles. The turning angle is bigger, and the value of h_{angle} is smaller. The heuristic function considers both the distance and turning factors so that the obtained optimal path should be a short path with few turns. In order to extend the search region in the initial stage the artificial ants choose next points randomly with big probability and use the transition probability calculated using Eq. (4) and the roulette method to choose the next points with small probability. In the initial stage the artificial ants will mainly focus on extending the searching region and the local minimum problem is alleviated. In the medium stage the artificial ants use the transition probability calculated using Eq. (4) and the roulette method to choose the next points which makes the system operate stably. In the end stage most of the artificial ants choose the next points with high transition probability and only few artificial ants use the transition probability calculated using (4) and the roulette method to choose the next points. This will make the system converge fast.

(3) Update the pheromone value using Eq. (2) and (3).

(4) Calculate the length of the paths searched in this iteration considering the turning factors. Find the best path and store every point of this path.

(5) Number of Iterations $NC = NC + 1$. If $NC < NC_{max}$ empty the taboo collection and go to step 2 continuously. If $NC = NC_{max}$ end the searching process and output the searched optimal path.

4. Simulation

In this paper MATLAB is used to realize the path planning simulation using the proposed modified ant colony algorithm. The operation environment is expressed by a 20×20 grid map. The number of the artificial ants is set as $m = 25$, maximum iterations $NC_{max} = 200$, $\alpha = 1$, $\beta = 6$, $\rho = 0.2$, and $Q = 800$. When the turning angles are 0° , 45° , 90° , 135° and 180° , the corresponding turning operators are 1, 0.9, 0.7, 0.5 and 0.1, respectively. Figs. 3 and 4 show the path planning results by using the modified ant colony algorithm. Here the dash areas are obstacles and arrows indicate the searched optimal path. From Figs. 3 and 4 it can be seen that in a known static environment the proposed modified ant colony algorithm can provide the mobile robot an optimal path. In the optimal process since the turning factor is considered the turning positions of the obtained path are few.

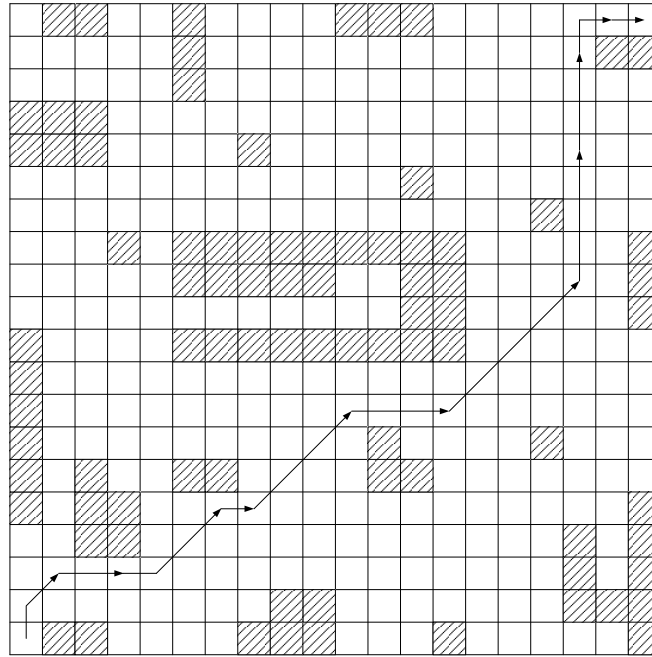


Figure 3. Path Planning Results 1 by Using the Modified Ant Colony Algorithm

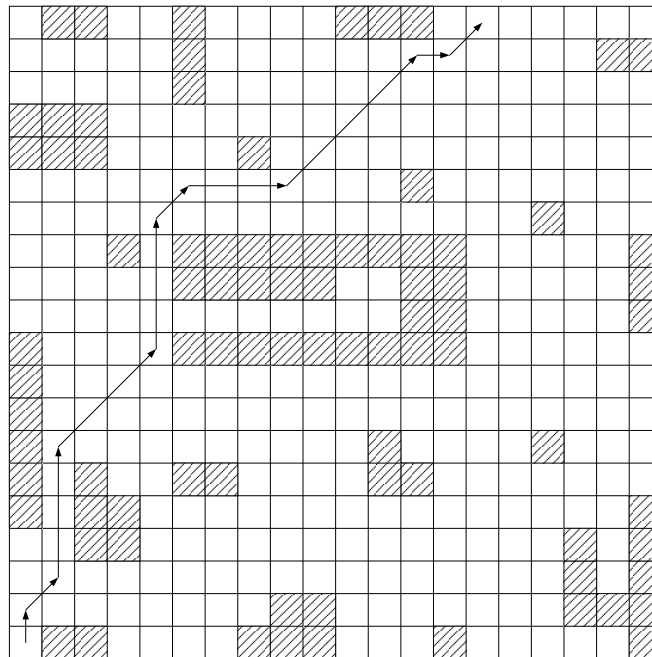


Figure 4. Path Planning Results 2 by Using the Modified Ant Colony Algorithm

5. Conclusion

This paper presents a modified ant colony algorithm for mobile robots path planning in a static environment. The target of mobile robots path planning is to find an obstacle-free shortest path from the start point to the end point. Ant colony algorithm is a frequently used swarm intelligent optimization method in various

applications. It has distributed calculation, self-organization and positive feedback advantages. But the convergence velocity is slow and the algorithm is trapped in local optimal result easily. The stagnation happens when there is concave shape in the environment. Sometimes the optimal path obtained by ant colony algorithm has many turns. In this paper we proposed a modified ant colony algorithm focusing on the four problems aforementioned. By considering the turning factor in the optimal process, the modified algorithm can provide the optimal path with few turning positions. In the initial stage of the algorithm the searching region is extended to alleviate the local minimum problem, and in the end stage the modified algorithm can converge fast. Simulation is conducted to verify the proposed modified ant colony algorithm. Good path planning results show the feasibility of the proposed algorithm for mobile robot path planning application.

Acknowledgments

This paper is supported by Jiangsu Province University Natural Science Research Project (13KJB510003), Qing Lan Project, and Jiangsu Institute of Marine Resources Open Fund Project (JSIMR201407).

References

- [1] R. Kala, A. Shukla and R. Tiwari, "Robotic Path Planning in Static Environment Using Hierarchical Multi-Neuron Heuristic Search and Probability Based Fitness", *Neurocomputing*, vol. 74, no. 14-15, (2011), pp. 2314-2335.
- [2] Y. Zhang, D. Gong and J. Zhang, "Robot Path Planning in Uncertain Environment using Multi-Objective Particle Swarm Optimization", *Neurocomputing*, vol. 103, (2013), pp. 172-185.
- [3] M. Oscar, O. R. Ulises and S. Roberto, "Path Planning for Mobile Robots using Bacterial Potential Field for Avoiding Static and Dynamic Obstacles", *Expert Systems with Applications*, vol. 42, no. 12, (2015), pp. 5177-5191.
- [4] A. T. Rashid, A. A. Ali, M. Frasca and L. Fortuna, "Path Planning with Obstacle Avoidance Based on Visibility Binary Tree Algorithm", *Robotics and Autonomous Systems*, vol. 61, no. 12, (2013), pp. 1440-1449.
- [5] F. Duchon, A. Babinec, M. Kajan, P. Beno, M. Florek, T. Fico and L. Jurišica, "Path Planning with Modified A Star Algorithm for A Mobile Robot", *Procedia Engineering*, vol. 96, (2014), pp. 59-69.
- [6] A. Vahid and S. Hamed, "Dual Hierarchical Genetic-Optimal Control: A New Global Optimal Path Planning Method for Robots", *Journal of Manufacturing Systems*, vol. 33, no. 1, (2014), pp. 139-148.
- [7] D. Marija and P. Ivan, "Two-Way D* Algorithm for Path Planning and Replanning", *Robotics and Autonomous Systems*, vol. 59, no. 5, (2011), pp. 329-342.
- [8] M. Hawa, "Light-Assisted A* Path Planning", *Engineering Applications of Artificial Intelligence*, vol. 26, no. 2, (2013), pp. 888-898.
- [9] S. S. Ge, and Y. J. Cui, "New Potential Functions for Mobile Robot Path Planning", *IEEE Transactions on Robotics and Automation*, vol. 16, no. 5, (2000), pp. 615-620.
- [10] Q. Li, C. Zhang, C. Han, Y. Xu, Y. Yin and W. Zhang, "Path Planning Based on Fuzzy Logic Algorithm for Mobile Robots in Static Environment", *Proceedings of the 25th Chinese Control and Decision Conference*, Guiyang, China, (2013) May 25-27.
- [11] Q. Hong, S. X. Yang, A. R. Willms and Y. Zhang, "Real-Time Robot Path Planning Based on A Modified Pulse-Coupled Neural Network Model", *IEEE Transactions on Neural Networks*, vol. 20, no. 11, (2009), pp. 1724-1739.
- [12] J. Espelosin, L. Acosta and D. Alonso, "Path Planning Approach Based on Flock Dynamics of Moving Particles", *Applied Soft Computing*, vol. 13, no. 4, (2013), pp. 2159-2170.
- [13] A. C. C. Marco, A. R. Victor and H. H. B. Uriel, "Mobile Robot Path Planning Using Artificial Bee Colony and Evolutionary Programming", *Applied Soft Computing*, vol. 30, (2015), pp. 319-328.
- [14] K. Ioannidis, G. C. Sirakoulis and I. Andreadis, "Cellular Ants: A Method to Create Collision Free Trajectories for A Cooperative Robot Team", *Robotics and Autonomous Systems*, vol. 59, no. 2, (2011), pp. 113-127.
- [15] Q. Zhu, J. Hu, W. Cai and L. Henschen, "A New Robot Navigation Algorithm for Dynamic Unknown Environments Based on Dynamic Path Re-Computation and An Improved Scout Ant Algorithm", *Applied Soft Computing*, vol. 11, no. 8, (2011), pp. 4667-4676.
- [16] D. Sudholt and C. Thyssen, "Running Time Analysis of Ant Colony Optimization for Shortest Path Problems", *Journal of Discrete Algorithms*, vol. 10, (2012), pp. 165-180.

- [17] I. Châari, A. Koubâa, H. Bennaceur, A. Ammar, S. Trigui, M. Tounsi, E. Shakshuki and H. Youssef, "On the Adequacy of Tabu Search for Global Robot Path Planning Problem in Grid Environments", *Procedia Computer Science*, vol. 32, (2014), pp. 604-613.
- [18] G. Tan, H. He and A. Sloman, "Ant Colony System Algorithm for Real-Time Globally Optimal Path Planning of Mobile Robots", *Acta Automatica Sinica*, vol. 33, no. 3, (2007), pp. 279-285.

Authors



Yuanliang Zhang, he graduated from Tsinghua University, China in 2001. He received the M. Sc. and Ph.D. degrees from Chonbuk National University, Korea in 2006 and 2009, respectively. He is currently a postdoctor at School of Automation, Southeast University and an associate professor at Jiangsu Institute of Marine Resources, Huaihai Institute of Technology, China. His research interests include control system and robotics.



Cheng Chen, he graduated from Nanjing Institute of Technology, China in 2014. He is currently a master candidate at Jiangsu Institute of Marine Resources, Huaihai Institute of Technology, China. His research interests include ant colony algorithm and robotics.



Qing Liu, he graduated from Huaihai Institute of Technology, China in 2005. She received the M. Sc. Degree from Nanjing University of Science and Technology, China in 2014. She is currently a lecturer at Jiangsu Institute of Marine Resources, Huaihai Institute of Technology, China. Her research interests include control system and robotics.