2008 8th IEEE-RAS International Conference on Humanoid Robots
December 1 ~ 3, 2008 / Daejeon, Korea

WP2-24

Improved Ant Colony Optimization Algorithm by Potential Field Concept for Optimal Path Planning

Joon-Woo Lee [1], Jeong-Jung Kim [2], Byoung-Suk Choi [3] and Ju-Jang Lee [4]

[1] *Korea Advanced Institute of Science and Technology, South Korea* good791@kaist.ac.kr

[2] *Korea Advanced Institute of Science and Technology, South Korea* rightcore@kaist.ac.kr

[3] *Korea Advanced Institute of Science and Technology, South Korea* bs_choi@kaist.ac.kr

[4] *Korea Advanced Institute of Science and Technology, South Korea* jjlee@ee.kaist.ac.kr

***Abstract*— In this paper, an improved Ant Colony Optimization (ACO) algorithm is proposed to solve path planning problems. These problems are to find a collision-free and optimal path from a start point to a goal point in environment of known obstacles. There are many ACO algorithm for path planning. However, it take a lot of time to get the solution and it is not to easy to obtain the optimal path every time. It is also difficult to apply to the complex and big size maps. Therefore, we study to solve these problems using the ACO algorithm improved by potential field scheme. We also propose that control parameters of the ACO algorithm are changed to converge into the optimal solution rapidly when a certain number of iterations have been reached. To improve the performance of ACO algorithm, we use a ranking selection method for pheromone update. In the simulation, we apply the proposed ACO algorithm to general path planning problems. At the last, we compare the performance with the conventional ACO algorithm.**

I. INTRODUCTION

The path planning problem is to find a collision-free and optimal path for a mobile robot from a start point to a goal point in the given environment. Some studies use the most widely known Genetic Algorithms (GA) to solve the problem [1]. In addition, there are many algorithms for path planning, such as the grid-based A* (A-star) algorithm [2], Artificial Potential Field (APF) method [3], Fuzzy Logic (FL) [4], Neural Networks (NN) [5] and so on.

In the path planning problem, we need to express the given environment as a considerable type of representation. There are generally four different types of representation [6]. Among them, the composite-space map method is a very efficient and widely used. In the composite-space map method, the space is discretized into a grid of rectangular cells (or voxels) and each cell is marked as an obstacle or a non-obstacle. Eventually, the path is represented as a consecutive sequence of cells from a start point cell to a goal point cell. The complexity of the scene is expressed by the number of the cells n in the map. There are many algorithms using the composite-space map method such as A* algorithm, ACO algorithm, GA and so on.

In this paper, we use the ACO algorithm with the composite-space map method. The ACO algorithm has been successfully applied in the optimization problems such as Traveling Salesman Problem (TSP) [7] and Quadratic Assignment Problem (QAP) [9]. Of course, there are also many ACO algorithm for path planning [?]. However, it take a lot of time to get the solution and it is not to easy to obtain the optimal path every time. It is also difficult to apply to the complex and big size maps. The proposed ACO algorithm in this paper differs from the conventional ACO algorithm for the path planning in four respects. First, the improved ACO algorithm uses the potential field instead of the distance used for the ant move rule in the existing ACO algorithm. Also, this potential field is used for making the initial pheromone field. Second, when the pheromone field updates, path is graded according to the cost (i.e. shortest distance and shortest step). Third, we uses the varying control parameters (i.e. α and β) to converge into the optimal solution rapidly when a certain number of iterations have been reached. Finally, we don't use generation concept of a conventional ACO algorithm to reduce the time required to solve problems.

The reminder of this paper is organized as follows: In the Section II and Section III, the conventional ACO algorithm and the potential field method are introduced respectively. In the Section IV, the improved ACO algorithm is described in detail. In the Section V, some experiments are performed and the results are shown by figures and tables. After that, we conclude this paper in the Section VI.

II. ANT COLONY ALGORITHM

The ACO algorithm is based on behavior of real ants. While moving, ants finding food deposit pheromone on the way to their nest and the others follow pheromone deposited by other ants before. As time goes on, pheromone is evaporated for new possibility and ants cooperate to choose a path with heavy laid pheromone. On this wise, they can search the shortest path from their nest to food source with only pheromone information. The ACO algorithm has a parallel architecture and a positive feedback loop mechanism.

The basic mathematical model of ACO algorithm has been first applied to the TSP [7]. The TSP is the problem to find the shortest path that traverses all cities exactly once and return to the start city.

In the TSP used the ACO algorithm, there are n cities and the total number of ants is M. Each ant choose the next city to move in accordance with the distance of between the current city and the next city and the intensity of pheromone. In the basic ACO algorithm model, it is the transition probability

978-1-4244-2822-9/08/$25.00 ©2008 IEEE

from city i to city j for the k-th ant as:

$$p_{ij}^k(t) = \begin{cases} \dfrac{[\tau_{ij}(t)]^\alpha [\eta_{ij}]^\beta}{\sum\limits_{k \in allowed_k} [\tau_{ik}(t)]^\alpha [\eta_{ik}]^\beta} & \text{if } j \in allowed_k \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Where $allowed_k = \{N - tabu_k\}$ or this is the set of remainder cities except that once selected among N cities. α and β are constant parameters that determine the relative influence of the pheromone and the heuristic on the decision of the ant. η_{ij} is the heuristic desirability and $\eta_{ij} = 1/d_{ij}$ where d_{ij} is the Euclidean length of path between city i and city j. τ_{ij} is the amount of pheromone trail on edge (i, j). After the ants in the algorithm ended their tours, the pheromone trail amount τ_{ij} value of every edge (i, j) are updated according to the following formula:

$$\tau_{ij}(t+n) = \rho \cdot \tau_{ij}(t) + \Delta\tau_{ij} \tag{2}$$

Where ρ is the local pheromone decay parameter and $\rho \subset (0, 1)$. The pheromone is evaporated as time goes on.

$$\Delta\tau_{ij} = \sum_{k=1}^{m} \Delta\tau_{ij}^k \tag{3}$$

Where $\Delta\tau_{ij}^k$ is the amount of per unit length of pheromone trail laid on edge (i, j) by the k-th ant between time t and time $t+n$. In the general ant-cycle model, it is given by:

$$\Delta\tau_{ij} = \begin{cases} \dfrac{Q}{L_k} & \text{if } k\text{−th ant uses edge } (i,j) \text{ in its tour} \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

Where Q is a constant and L_k is the length of path found by the k-th ant.

This iteration process goes on until satisfying some termination conditions, e.g. a certain number of iterations have been achieved or the best path didn't change for several iterations. [10].

III. Potential Field Method

In the potential field method, the robot is treated as a point moving on an artificial potential field $U(q)$. The movement of the robot is just as a ball would roll downhill. The goal point acts as an attractive force on the robot and the known obstacles act as repulsive forces. The superposition of all forces impacts on the robot. Therefore, an artificial potential field guides the robot toward the goal point while simultaneously avoiding obstacles.

If we assume a differentiable potential field function $U(q)$, the related artificial force $F(q)$ acting at the position $q = (x, y)$ is found as:

$$F(q) = -\Delta U(q) \tag{5}$$

Where $\Delta U(q)$ denotes the gradient vector of U at position q,

$$\Delta U(q) = \begin{bmatrix} \dfrac{\partial U}{\partial x} \\ \dfrac{\partial U}{\partial y} \end{bmatrix} \tag{6}$$

The potential field is computed as the sum of the attractive field of the goal point and the repulsive fields of the obstacles:

$$U(q) = U_{att}(q) + U_{rep}(q) \tag{7}$$

Similarly, the forces can also be represented:

$$\begin{aligned} F(q) &= F_{att}(q) - F_{rep}(q) \\ &= -\Delta U_{att}(q) - \Delta U_{rep}(q) \end{aligned} \tag{8}$$

A. Attractive Potential

An attractive potential can, for example, be defined as a parabolic function.

$$U_{att}(q) = \frac{1}{2} k_{att} \cdot d_{goal}^2(q) \tag{9}$$

Where k_{att} is a positive scaling factor and $d_{goal}(q)$ denotes the Euclidean distance $\|q - q_{goal}\|$. This attractive potential is differentiable and then the attractive force $F_{att}(q)$ is computed as:

$$\begin{aligned} F_{att}(q) &= -\Delta U_{att}(q) \\ &= -k_{att} \cdot d_{goal}(q) \Delta d_{goal}(q) \\ &= -k_{att} \cdot (q - q_{goal}) \end{aligned} \tag{10}$$

that converges linearly toward 0 as the robot reaches the goal.

B. Repulsive Potential

One example of a repulsive field is defined as:

$$U_{rep}(q) = \begin{cases} \frac{1}{2} k_{rep} \left(\dfrac{1}{d(q)} - \dfrac{1}{d_0} \right)^2 & \text{if } d(q) \le d_0 \\ 0 & \text{if } d(q) \ge d_0 \end{cases} \tag{11}$$

Where k_{rep} is a scaling factor, $d(q)$ is the minimal distance form q to the obstacle and d_0 is the distance of influence of the obstacle. The repulsive potential function U_{rep} is positive or zero and tends to infinity as q gets closer to the obstacle.

If the object boundary is convex and piecewise differentiable, $d(q)$ is differentiable everywhere in the free configuration space. Then, the repulsive force F_{rep} is computed as:

$$\begin{aligned} F_{rep}(q) &= -\Delta U_{rep}(q) \\ &= \begin{cases} k_{rep} \left(\dfrac{1}{d(q)} - \dfrac{1}{d_0} \right) \dfrac{1}{d^2(q)} \dfrac{q - q_{obstacle}}{d(q)} & \text{if } d(q) \le d_0 \\ 0 & \text{if } d(q) \ge d_0 \end{cases} \end{aligned} \tag{12}$$

Consequently, the resulting force $F(q) = F_{att}(q) + F_{rep}(q)$ acts on the robot and moves the robot toward the goal point while simultaneously avoiding obstacles [11].

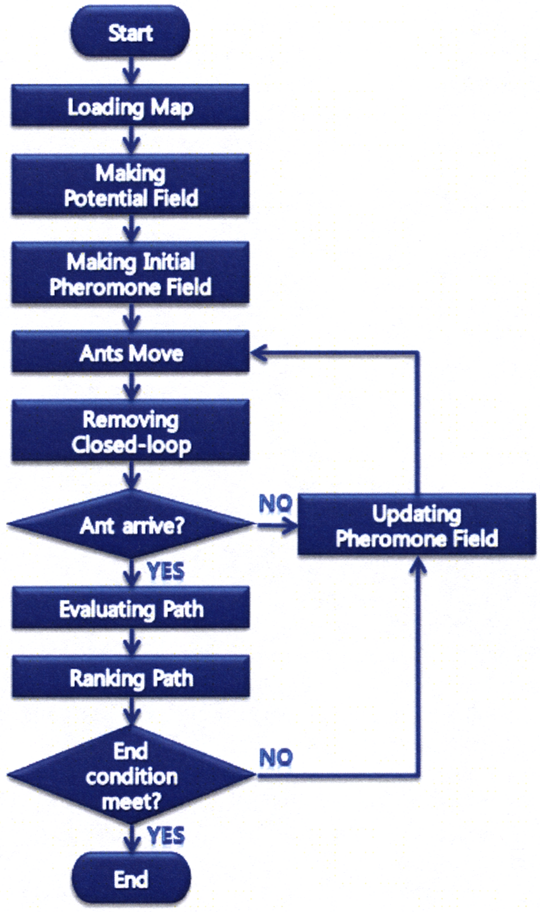

Fig. 1. The flowchart of the improved ACO algorithm

IV. Improved Ant Colony Algorithm

Fig. 1 shows the flowchart of the improved ACO algorithm. The improved ACO algorithm differ from the conventional ACO algorithm for the path planning in four respects as above stated in Section I. This four points of difference are described in detail as follow subsections. We suppose the several things in advance. First, ant move one cell per unit time according to the ant move rule. Second, all ants remember their path passed before from a start point to a goal point. Third, the path of ant arrived at the goal point is only used for updating pheromone field.

A. Ant Move Rule

In the improved ACO algorithm, we use the potential value $U(q)$ instead of η in Eq. (1). Where the potential value $U(q)$ differ from that of the conventional potential field. In the potnetial field, the start point makes an attractive field $U_{att}(q)$ as Fig. 2. Because η is the reciprocal of the Euclidean distance d from the current point to the goal point and the closer the goal point is, the lager η is. The reason why we use $U(q)$ instead of η is that $U(q)$ has a great attraction to the goal point stronger than η. The closer the goal point is, the steeper $U(q)$ has as Fig. 2. Accordingly, the use of the potential value $U(q)$ is better effect to converge into the optimal path rapidly.

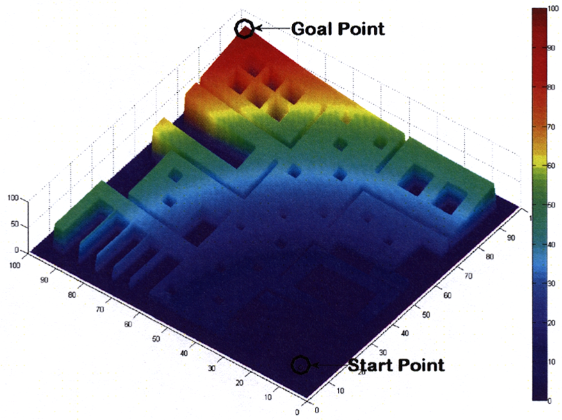

Fig. 2. The potential field of Map No.5

In the improved ACO algorithm, it is the transition probability from cell i to cell j for the k-th ant as:

$$p_{ij}^{k}(t) = \frac{[\tau_{ij}(t)]^{\alpha}[U_{ij}(q)]^{\beta}}{\sum\limits_{k \in allowed_k} [\tau_{ik}(t)]^{\alpha}[U_{ik}(q)]^{\beta}} \quad \text{if } j \in allowed_k \tag{13}$$

Where $allowed_k = \{k \mid k =$(8 cells around the current cell $i)\}$.

B. Pheromone Update Rule

The ACO algorithm has a pheromone field to store the pheromone information. In the conventional ACO algorithm, the initial pheromone quantity of the pheromone field is randomly determined for various possibility. However, this method waste the initial time. In point of the necessary time, it's an important respect of how soon the first ant arrived at the goal point because of the nature of ACO algorithm. The ACO algorithm can be operate actively with much pheromone connecting between the start point and the goal point. In this respect, the pheromone of the first ant arrived is their first and important source that certainly guides them toward the goal point. Under the known environment, it is most likely to find path toward the goal point as the ants move in the direction of the goal point in an initial state. Therefore, we proposed that the sum of the potential field and the random pheromone field as the initial pheromone field. In this proposed method, while the influence of the potential field is strengthened, the influence of the random selection isn't neglected. The initial pheromone field are computed according to the following formula:

$$\tau_{sj}(0) = k_{mix} \cdot \tau_{random} + (1 - k_{mix}) \cdot \tau_{potential} \tag{14}$$

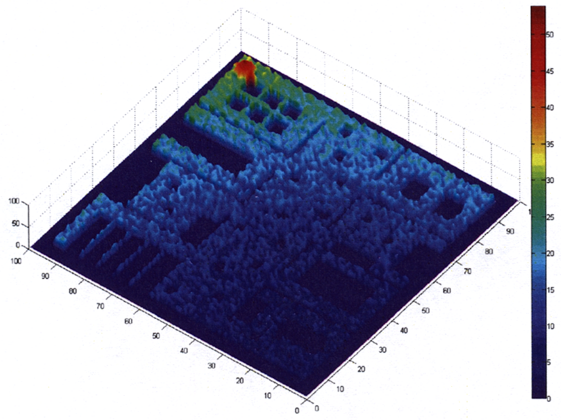

Fig. 3. The initial pheromone field of Map No.5

Where k_{mix} is the mixing constant and if this value is large, the influence of the random selection is strengthened. s is the start point cell. Fig. 3 shows the initial pheromone field with $k_{mix} = 0.3$.

In the improved ACO algorithm, an pheromone update rule also is modified about the several things. First, we use a ranking selection method for the pheromone update of the valid paths. We make a ranking list that has as many as 30 percent of the number of ants. Every time ant arrived at the goal point, the value of the cost function determines the rank of the path and the rank determines the quantity of the pheromone laid on the unit length of the path. This way we can keep better paths for a long time. The cost function for ranking is defined as:

$$(Cost) = (Path\ \ Length) \times (Number\ \ of\ \ cells) \quad (15)$$

The reason why *Number of cells* is used is that the path has either a diagonal path and a vertical or horizontal path.

Second, to converge into the optimal solution rapidly, we limit the decayed quantity of the pheromone and if the quantity of the pheromone converge into a certain value, we reset the pheromone field according to several conditions. In the first time, we update the converged pheromone field randomly for various possibility until all items in the ranking list are newly organized. Then, the converged pheromone field is updated by the 10 percent path of the ranking list until the items of the ranking list are newly changed as many as the number of the ants. After that, the converged pheromone field is updated by all path in the ranking list. The length variation of paths in the ranking list reduces as time goes on. Therefore, the above method is efficient.

C. Varying Control Parameters, α and β

In this section, we propose that the control parameters of the ACO algorithm is changed according to several conditions. Where the conditions is the same condition of the converged pheromone field reset stated in the Subsection IV-B. In the beginning, the influence of the potential field is strengthened (i.e. $\alpha < \beta$) to search a various path. Then, the influence of the pheromone field is strengthened (i.e. $\alpha > \beta$) because there are many valid paths in the ranking list.

D. Single Generation in Colony

Finally, we don't use the generation concept in the ACO algorithm. Because the improved ACO algorithm use a single generation, there is a different feature. If an ant is met the following two conditions, the position of the ant is reset. First, the path length is longer than the longest path in the ranking list. Second, an ant arrives at the goal point. The reset ants restart at the start point. The reason why these condition is used is to reduce the unnecessary exploration and exploitation for improving the efficiency.

V. EXPERIMENTS

In this section, we show the performance of the improved ACO algorithm as applying to the path planning problems and compare the performance with the conventional ACO algorithm. We use the language of MATLAB for the simulation. All simulations are implemented on PC that has a Intel Duo Core CPU @ 1.86GHz and 2GB of RAM under Windows XP.

The control parameters of the improved ACO algorithm are set the following as: The number of ants $M = 500$, $\alpha = 1$ and $\beta = 2$ at the initial condition, $\alpha = 2$ and $\beta = 1$ at the second condition, $\alpha = 3$ and $\beta = 1$ at the third condition, $rho = 0.99$, The number of the path in the ranking list: 100 and $k_{mix} = 0.3$.

In the case of the conventional ACO algorithm, the control parameters are set the following as: The number of ants $M = 500$, $\alpha = 1$ and $\beta = 2$ all the experiment time, $rho = 0.99$, The number of the path in the ranking list: 100.

We show the performance of the improved ACO algorithm and the conventional ACO algorithm with maps in Fig 4 Fig 8. We use 5 maps of the gird-based environment: Map No.1: 25×25, Map No.2: 50×44, Map No.3, Map No.4 and Map No.5: 100×100. In the map, the red square is the start point and the blue square is the goal point.

In the Fig 4-8 and Table I-II, there are the results of the simulation. The red line is the optimal path searched using the improved ACO algorithm and the blue line is that using the conventional ACO algorithm in the figures. The whole of simulation results are acceptable to the optimal paths without reference to the complexity of the map. In respect of the time, the search time of the proposed ACO algorithm is shorter than that of the conventional ACO algorithm. In case of the conventional ACO algorithm, it's hard and take a long time to find the path including the reverse direction as Map No.1 in Fig. 4. The big size and complex maps as Fig. 7 and Fig. 8 aren't also an easy problems and take a long time. In respect of the quality of path, the path searched using the improved ACO algorithm is better and shorter than another.

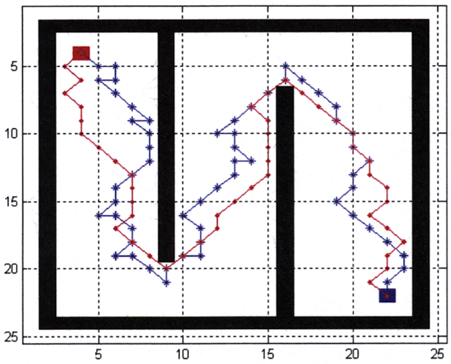

Fig. 4. The solution path of Map No.1

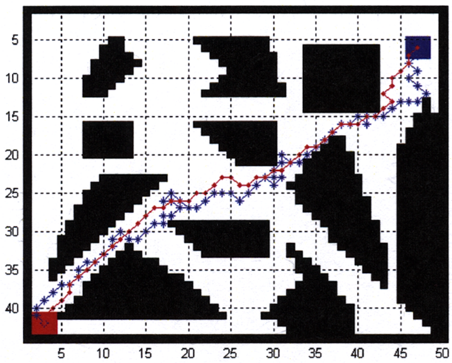

Fig. 5. The solution path of Map No.2

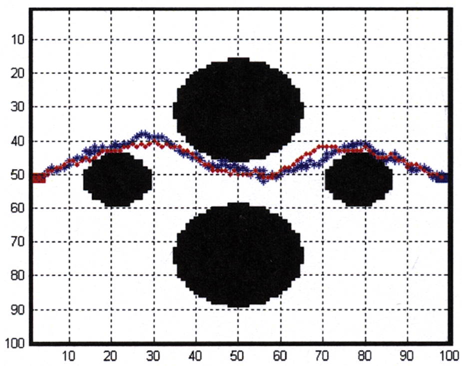

Fig. 6. The solution path of Map No.3

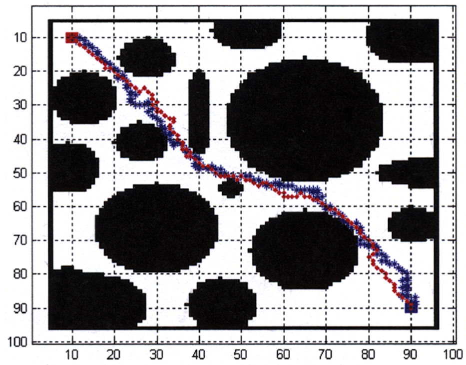

Fig. 7. The solution path of Map No.4

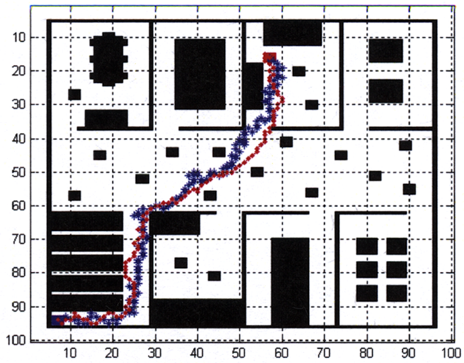

Fig. 8. The solution path of Map No.5

TABLE I
SIMULATION RESULTS(CONVENTIONAL ACO)

	Path length	Number of cells	Search time(sec)
Map No.1	72.0833	59	13326
Map No.2	83.3970	67	3926
Map No.3	154.2376	125	16552
Map No.4	175.2376	146	17953
Map No.5	168.6518	139	22641

TABLE II
SIMULATION RESULTS(IMPROVED ACO)

	Path length	Number of cells	Search time(sec)
Map No.1	59.2548	47	602
Map No.2	64.0833	51	274
Map No.3	123.6102	101	1063
Map No.4	135.8234	107	1201
Map No.5	131.6812	107	1593

VI. CONCLUSIONS

In this paper, we proposed an improved ACO algorithm by potential field concept for optimal path planning. The conventional ACO algorithm has applied to only a simple and small size map. However, the proposed algorithm can be applied to bigger and more complex map. The quality of the path is also acceptable. But there are some defects. It is hard to get a smooth path for physical mobile robot and it takes a long time to search the optimal path compared to other path planning methods. In the future, we need to reduce search time and make the path smoothly. In addition, we need to compare with other optimization algorithms for path planning in detail.

ACKNOWLEDGMENT

Authors are gratefully acknowledging the financial support by Agency for Defence Development and by UTRC(Unmanned Technology Research Center), Korea Advanced Institute of Science and Technology.

REFERENCES

[1] W.-G. Han, S.-M. Baek and T.-Y. Kuc, "GA based on-line path planning of mobile robots playing soccer games," Proceedings of the 40th Midwest Symposium on Circuits and Systems, vol.1, pp. 522-525, Aug. 1997.

[2] Warren, C.W. "Fast path planning using modified A* method," Proceedings of the 1993 IEEE International Conference on Robotics and Automation, vol.2, pp. 662-667, May 1993.

[3] F. Janabi-Sharifi, F. Vinke and D. Vinke, "Robot path planning by integrating the artificial potential field approach with simulated annealing," Proceedings of the 1993 IEEE International Conference on Systems, Man and Cybernetics, vol.2, pp. 282-287, Oct. 1993.

[4] M.-S. Lee, M.-J. Jung and J.-H. Kim, "Evolutionary programming-based fuzzy logic path planner and follower for mobile robots," Proceedings of the 2000 IEEE Congress on Evolutionary Computation, vol.1, pp. 139-144, July 2000.

[5] R Glasius, A Komoda, SCAM Gielen, "Neural Network Dynamics for Path Planning and Obstacle Avoidance," Neural Networks, vol.8, no.1, pp. 125-133, 1995.

[6] Bandi, s., Talmann, D., "Space discretization for efficient human navigation," Computer Graphic Forums, vol.17, no.3, pp. 195-206, 1998.

[7] M. Dorigo and L.M. Gambardella, "Ant colonies for the travelling salesman problem," BioSystems, vol.43, no.2, pp 73-81, July 1997

[8] Tsutsui, S. and Lichi Liu, "Solving quadratic assignment problems with the cunning ant system," IEEE Congress on Evolutionary computation 2007, pp. 173-179, Sept. 2007.

[9] N. H. Viet, N. A. V, S.-G. Lee and T.-C. Chung, "Obstacle Avoidance Path Planning for Mobile Robot Based on Multi Colony Ant Algorithm," IEEE International Conference on Advances in Computer-Human Interaction, pp. 285-289, Feb. 2008.

[10] H. B. Duan, G. J. Ma and S. Q. Liu, "Experimental study of the adjustable parameters in basic ant colony optimization algorithm," IEEE Congress on Evolutionary computation 2007, pp. 149-156, Sept. 2007.

[11] R. Siegwart and Illah R. Nourbakhsh, "Introduction to Autonomous Mobile Robots," 1st ed., The MIT press, Cambridge, Massachusetts, pp. 267-270, April 2004.