ISSN 1975-8359(Print) / ISSN 2287-4364(Online)
The Transactions of the Korean Institute of Electrical Engineers Vol. 66, No. 1, pp. 114～120, 2017
http://doi.org/10.5370/KIEE.2017.66.1.114

미지 동적 환경에서 다중 이동로봇의 GA-Fuzzy 기반 자율항법

GA-Fuzzy based Navigation of Multiple Mobile Robots in Unknown Dynamic Environments

조 연* · 이 홍 규†
(Ran Zhao · Hong-Kyu Lee)

Abstract - The work present in this paper deals with a navigation problem for multiple mobile robots in unknown indoor environments. The environments are completely unknown to the robots; thus, proximity sensors installed on the robots' bodies must be used to detect information about the surroundings. The environments simulated in this work are dynamic ones which contain not only static but also moving obstacles. In order to guide the robot to move along a collision-free path and reach the goal, this paper presented a navigation method based on fuzzy approach. Then genetic algorithms were applied to optimize the membership functions and rules of the fuzzy controller. The simulation results verified that the proposed method effectively addresses the mobile robot navigation problem.

Key Words : Robot navigation, Fuzzy, Dynamic environments, Genetic algorithm, Multiple robots

1. Introduction

As an important branch subject of robotics, autonomic mobile robot technology has a long history and will be used widely in future. Mobile robots have vast application prospects in areas including space exploring, factory automation, mining, eliminating dangerous situation, military and service, which can economize the labor force to be engaged in other aspects. In these applications, the navigation problem of mobile robots is one of the most popular issues. Thus, how to detect the surrounding information and finding a safe path for the robot is the first condition of success. In an environment with obstacles, the navigation of a robot together with its path planning is to find the collision-free path from a starting location to a target location.

In the past few decades, several methods have been suggested to solve the navigation problem for mobile robot. Such as neural network [1], genetic algorithm [2], artificial vision method [3] and PID control [4] in static environment. The methods above are usually used in global path planning and hardly be used in real-time control. There are also many traditional methods designed for global path planning have also been extended to local path planning, such as potential field method [5], roadmap method [6] and rolling window method [7], etc.

The fuzzy logic for the navigation and obstacle avoidance problem of mobile robots has also been studied by researchers for years. In [8], researchers suggested a minimum hazard approach based on fuzzy logic controller, by which the goal-oriented robot can escape from the local minimum location during the navigation process in unknown environment. Nevertheless, the method mentioned in [9] is combined the fuzzy logic with simulated annealing. By this method, the mobile robot shows good performance in obstacle avoidance.

Navigation for multiple mobile robots is another important issue for researchers. Zhao [10] developed a new finite-time synchronized approach for multiple mobile robot formation control, based on a terminal sliding mode control principle and system synchronization theory. Zhong [11] established a new Velocity-Change-Space method that performs dynamic motion planning by analyzing changes in the speed and direction of the robot's velocity. However, robots must measure the size, positions, and velocities of obstacles and other robots online. Most of above researches need a global perspective, such as central server, global image or part of environment information. Therefore, there is an increasing need to study the proposed problem in completely unknown environments. Otherwise, one of the

† Corresponding Author : Dept. of Electrical Engineering, Korea University of Technology and Education, Korea.
E-mail: hongkyu@koreatech.ac.kr
* Dept. of Electrical Engineering, Korea University of Technology and Education, Korea.
Received : December 1, 2016; Accepted : December 16, 2016

most important questions is to establish high-efficiency membership functions and rules set of the fuzzy logic controller (FLC), which will influence the performance of the fuzzy inference. In this work we focus on the optimized fuzzy logic based navigation problem for multiple mobile robots in completely unknown dynamic environments. And the proposed fuzzy logic is evolved by Genetic Algorithm (GA).

The remainder of this paper is organized as follows: Section 2 presents the modeling of the robot and kinematic functions. Section 3 describes the navigation strategies and FLCs. Section 4 presents an optimization method for FLC by GA. In Section 5, we verify the effectiveness of the proposed method by simulation. Finally, conclusions and discussions are included in Section 6.

2. Problem formulation and working assumptions

The simulations in this work are mainly conducted with a classic wheeled mobile robot. This robot is supported by two DC motors, installed on the left and right wheels. Encoders on each wheel allow the robot to detect how far the wheel has traveled. Typically, a gyro sensor is also installed on the robot body; this sensor provides feedback information and can correct driving direction in practice. The structure of such a robot (with a circular shape) is shown in Fig. 1. In total, there are 16 proximity sensors (ultrasonic sensors) equably arranged around the robot body in equal intervals. The sensors, which are numbered from s_0 to s_{15}, are used to measure distances between the robot and surrounding obstacles or other robots.

The data that defines a path element can be denoted as a series of coordinate values defining points. We use the generalized coordinates $[x, y, \theta]^T$ to describe the configuration of every point. In this work, we assume that the robot moves on a flat surface; moreover, inertial effects are neglected. The robot is driven under conditions with no slippage; thus, it should follow the non-holonomic constraint as:

$$\dot{x}sin\theta - \dot{y}cos\theta = 0. \tag{1}$$

where θ is the angle between the robot's driving direction and x-axis. d is the distance between P_W and P_R, which respectively denote the center point between the two wheels, and the center of the robot. The kinematic functions of P_R can then be given as:

$$\begin{cases} \dot{x} = \dfrac{v_L + v_R}{2} cos\theta + \dot{\theta} d sin\theta \\ \dot{y} = \dfrac{v_L + v_R}{2} sin\theta + \dot{\theta} d cos\theta \\ \dot{\theta} = \dfrac{v_L - v_R}{L} \end{cases}, \tag{2}$$

where v_L and v_R denote the corresponding linear velocities of the left and right wheels, respectively, and L denotes the distance between the two wheels.

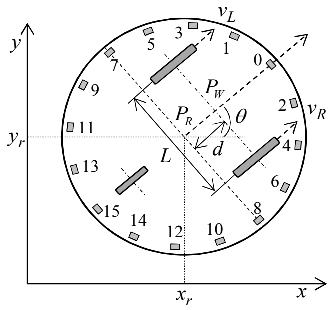

Fig. 1 Modeling of a wheel mobile robot

3. Fuzzy Logic Controllers

In order to drive the mobile robot real-timely in totally unknown environments, navigation is completely achieved in a reactive manner. The proposed approach here is based on fuzzy inference system and is inspired by human behaviors, which consists of danger judgment strategy and target positioning strategy.

3.1 Danger Judgment Strategy for Obstacle Avoidance

The "danger judgment strategy" developed here is used to avoid convex obstacles or dynamic ones. Firstly, at the start position, the robot moves ahead with the direction towards the target. Once the sensors installed on the robot body detect any objects (obstacles or other robots), this strategy is to be activated. It is obtained by means of a fuzzy logic controller (FLC1) and based on a set of rules which generated by human experience.

The sensors installed on the robot body are set to run once by every 0.1 second. Thus, in such a short time, we can simplify the relative velocity calculation as the following equation to obtain the acceptable approximate solutions:

$$v_{ro}^i \approx \frac{d_{t1}^i - d_{t2}^i}{\Delta t}, \tag{3}$$

where d^i_{t1} and d^i_{t2} are the distance measured by the i_{th} sensor at time t_1 and t_2. Here, the time interval Δt of t_1 and t_2 is 0.1 second. We use the expected time of collision to denote the danger coefficient, that is:

$$R^i_{dc} = d^i_{t2} / v^i_{ro}. \tag{4}$$

The smaller of the value of R^i_{dc} is, the more dangerous the robot will be. Specially, when the robot moves away from the obstacles, the value of R^i_{dc} is set as infinity. We denote that the danger coefficients of all the sensors are:

$$P_{dc} = \left\{ R^i_{dc} | 0 \le i \le 15 \right\}. \tag{5}$$

Here, the angle of the corresponding i_{th} sensor with the minimum danger coefficient is judged as the most dangerous direction of coming obstacles, and will be firstly dealt with. The method used for avoiding this dangerous collision is achieved by a fuzzy logic controller which is named as FLC1 in this section. Here, the minimum danger coefficient and the angle of the corresponding sensor are taken as the input variables. Moreover, the linear velocity of left and right wheels will be the output variables. Then the input variables can be described as:

$$R = \min(P_{dc}), \tag{6}$$

$$\Phi = \angle s_i, when\, R^i_{dc} = \min(P_{dc}), \tag{7}$$

where $-180^\circ \le \angle s_i \le 180^\circ$. For the fuzzification process, the input and output variables can be divided into several linguistic terms as follow:

S: small; M: medium; B: big; VB: very big;
RB: right back; R: right; RF: right front; F: front;
LF: left front; L: left; LB: left back; B: back;
Z: zero; NB: negative big; NM: negative medium;
NS: negative small; PB: positive big;
PM: positive medium; PS: positive small.

The process that transforms non-fuzzy input values into fuzzy values is achieved using membership functions that provide fuzzy terms with a definite meaning. The degree of membership of a no-fuzzy input value to a certain fuzzy set represents the confidence, expressed as a number from 0 to 1, that a particular value belongs to this fuzzy set [10]. In this work, we used the triangular membership functions. For the i_{th} linguistic term A_i, a triangular sharp can be defined by three points: $A_{i,l}$, $A_{i,c}$ and $A_{i,r}$, which means the left, center and right point. According to A_i together with the corresponding input x_i, the output value μ_i can be described as the following equation:

$$\mu_i = \begin{cases} \dfrac{(A_{i,c} - A_{i,r}) - (A_{i,c} - x_i)}{A_{i,c} - A_{i,l}}, A_{i,l} \le x_i \le A_{i,c} \\ \dfrac{(A_{i,r} - A_{i,c}) - (x_i - A_{i,c})}{A_{i,r} - A_{i,l}}, A_{i,c} \le x_i \le A_{i,l} \\ 0, \qquad others \end{cases}. \tag{8}$$

The membership functions for input and output variables of FLC1 are as shown in Fig. 2. As shown in this figure, when the main risk comes from left side, angle Φ is positive, otherwise it is negative.

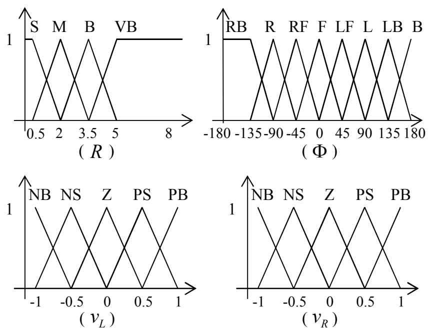

Fig. 2 Membership functions for input and output variables of FLC

The fuzzy inference procedure usually contains a set of rules that are used to appoint the desired control behavior. A rules set is a condition description taking the form of "IF…THEN…" rules. The rules set applied to FLC1 is shown in Table 1.

Table 1 Rules set for FLC1

v_L		Φ							
		RB	R	RF	F	LF	L	LB	B
R	S	PB	PB	NB	NB	NS	PB	PB	PS
	M	PB	PB	NB	NB	PB	PB	PS	PS
	B	PB	PB	NS	Z	PB	PB	PS	PS
	VB	PB	PB	PS	PS	PB	PB	PS	PS

v_R		Φ							
		RB	R	RF	F	LF	L	LB	B
R	S	PB	PB	NS	NS	NB	PB	PB	PB
	M	PS	PB	PB	Z	NB	PB	PB	PB
	B	PS	PB	PB	PB	NS	PB	PB	PB
	VB	PS	PB	PB	PB	PS	PB	PB	PB

3.2 Target Positioning Strategy

When there is no risk of collision, another strategy for target positioning is to be active. The new FLC which the target orientation process is achieved by is denoted as FLC2.

The schematic model of target positioning is as given in Fig. 3. In this figure, with the current position P_R, the robot has moved over the obstacle P_O. With the angle ψ_t between the line from the center of the robot to the target point and the x-axis, we denote that the angular difference is $\Psi = \theta - \psi_t$. Now, the obstacle is moving far away from the robot and poses no risk to the robot. Thus, the strategy of FLC2 is come into use. This controller is used to drive the robot to point to the target position. In the other words, it is used to reduce the angular difference Ψ which is one of the input variables of FLC2.

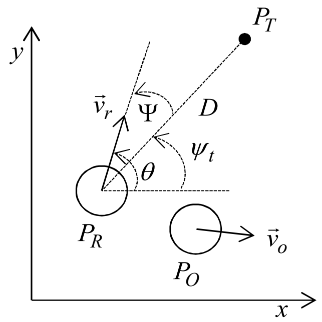

Fig. 3 Scheme of target positioning

Another constraint for FLC2 is the distance between the robot and its target, which is denoted with the multiple of robot's radius in this paper. Moreover, the output variables also are the linear velocities of left and right wheels. Similar with FLC1, the triangular type of membership functions are used for both input and output variables of FLC2.

4. Optimized FLC by GA

The rules set and membership functions discussed in section 3 were created all by human experience, which will influence the performance of a FLC. Thus, it is indispensable to adjust them to get a better performance. A genetic algorithm is employed here to do this work. GA is a search strategy based on models of evolution in nature. The optimization procedure of GA requires several steps which can be achieved in accordance with the following order. Because there is not much value to optimize the parameters of FLC2, this paper will focus on the optimization of the fuzzy controller of obstacle avoidance.

4.1 Chromosomes and initialization

A chromosome corresponds to a possible solution of the optimization problem and every chromosome is composed of several encoded genes. In order to encode a FLC, we integrated the proposed encoding procedure with both the membership functions and rules set of FLC1. We use A_i, B_i, C_i, D_i to denote the linguistic values of input and output variables, where i is the number of corresponding linguistic values. For the triangular membership functions every linguistic value can be described by three points. But in particular, the linguistic values on both ends of every membership function are described by two points. Because the endpoints are fixed to the end by experience. Hence, we can encode the membership functions which with the length of 58 in the way as shown in Fig. 4. Number 1 to 5 are used to encoding the linguistic values of the rules set. Thus, the rules set can be encoded as Table 2.

$$
\begin{array}{cc}
A_{1,c}, A_{1,r}, A_{2,l}, A_{2,c}, A_{2,r}, A_{3,l}, A_{3,c}, A_{3,r}, A_{4c}, A_{4,r}, & 10\text{bit} \\
+ & + \\
B_{1,c}, B_{1,r}, B_{2,l}, B_{2,c}, B_{2,r}, \ldots, B_{7,l}, B_{7,c}, B_{7,r}, B_{8c}, B_{8,r}, & 22\text{bit} \\
+ & + \\
C_{1,c}, C_{1,r}, C_{2,l}, C_{2,c}, C_{2,r}, \ldots, C_{4,l}, C_{4,c}, C_{4,r}, C_{5c}, C_{5,r}, & 13\text{bit} \\
+ & + \\
D_{1,c}, D_{1,r}, D_{2,l}, D_{2,c}, D_{2,r}, \ldots, D_{4,l}, D_{4,c}, D_{4,r}, D_{5c}, D_{5,r} & 13\text{bit}
\end{array}
$$

Fig. 4 Encoding for membership functions

Table 2 Encoding for membership functions of FLC1

v_L		Φ							
		RB	R	RF	F	LF	L	LB	B
R	S	5	5	1	1	2	5	5	4
	M	5	5	5	5	5	5	4	4
	B	5	5	2	3	5	5	4	4
	VB	5	5	4	4	5	5	4	4

v_R		Φ							
		RB	R	RF	F	LF	L	LB	B
R	S	5	5	2	2	1	5	5	5
	M	4	5	5	3	1	5	5	5
	B	4	5	5	5	2	5	5	5
	VB	4	5	5	5	4	5	5	5

GAs are efficient techniques for searching for global optimum solutions but may have premature convergence problems [12]. In our case, the number of original individuals of the first generation is initialized as 30. In order to improve the convergence rate of GA and avoid the premature convergence, we added three same individuals which generated by the input and out values and rules set discussed in the previous section. The rest 27 individuals are produced randomly with real number coding format.

4.2 Fitness functions

In our case, a feasible fitness function should include two elements, that is, efficiency and security. Thus, for the i_{th} individual, the running time T_i of the process from the start location to target location, the total path length L_i and the distance d_i between the robot body and the surrounding obstacles should be taken into account. The path length can be described by the following equation:

$$L_i(t) = \int_0^{T_i} dv \times dt \tag{9}$$

A penalty function is required to insure that the robot will not collision any obstacles. Denote that d_s is the safety distance, the penalty function can be designed as:

$$d_i = \begin{cases} 1, & d_i \le d_s \\ 0 & d_i > d_s \end{cases} \tag{10}$$

Then the fitness function can be written as:

$$F_i = \frac{1}{w_1 L_i + w_2 T_i + w_3 d_i} \tag{11}$$

where w_1, w_2, w_1 are weighting coefficients.

4.3 GA operators

In the proposed algorithm, we used three traditional GA operators: selection, crossover and mutation. Here the 'Stochastic Universal Sampling' (SUS) is used as the selection operator. At each generation, all the chromosomes are be updated based upon their fitness values. In other words, if a particular chromosome has better fitness than other chromosomes, then that particular chromosome is more likely to win the competition and clone itself. In addition, the 'Double-Point Crossover' operator and 'Real- valued Mutation' operator were used for crossover and mutation procedure in this work.

4.4 Optimization results

By following the above steps, now we can implement the optimization procedure. The optimization environments of robot moving space are same as the environments shown in Fig. 7, which contains one mobile robot, one static and five dynamic obstacles. Several essential parameters are defined as: crossover rate is 0.9, mutation rate is 0.01 and maximum generation is 500. Therefore, through the evolution, we can get the curve of the maximum and average fitness values in every generation as shown in Fig. 5. The optimized membership functions are shown in Fig. 6. Table 3 describes the evolved rules set from Table 1. It's important to note that when the optimized membership functions and rules set are applied to simulations, both of them must be applied simultaneously, because the optimized parameters of FLC are evolved together by every generation.

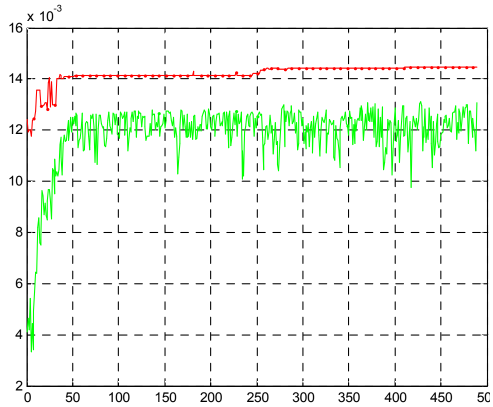

Fig. 5 Evolution of the maximum (up) and average (down) fitness values

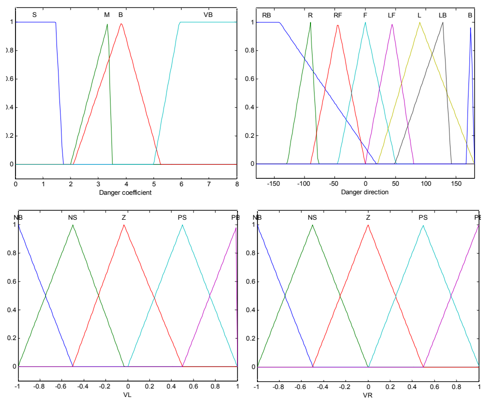

Fig. 6 Optimized membership functions of FLC1

Table 3 Evolved fuzzy rules set of FLC1

v_L		Φ							
		RB	R	RF	F	LF	L	LB	B
R	S	PB	PS	NB	NB	NS	PS	PS	PS
	M	RB	PB	NB	NB	PB	PB	PB	NS
	B	PB	PB	NS	Z	PB	PB	PS	PS
	VB	PB	PB	PB	PS	PB	PB	PB	PB

v_R		Φ							
		RB	R	RF	F	LF	L	LB	B
R	S	PB	PS	NS	NB	NB	PB	PB	PB
	M	RS	PB	PB	NB	NB	PB	PB	PB
	B	PS	PB	PB	PB	NB	PB	NB	NS
	VB	PS	PB	PB	PB	PB	Z	Z	NS

5. Simulation Results

In order to verify the effectiveness of the algorithm we have discussed, a series of simulations have been implemented with Matlab. The robot will move on a horizontal plane with the size of unit-length and the accelerated speed won't be considered. The results of one robot in the environments with 1 static and 5 dynamic obstacles are shown in Fig. 7. The dotted line denotes the trajectories of dynamic obstacles. Here, the dynamic obstacles, which with different radius, speed and moving directions, will continue moving back and forth during the entire process.

Compare the paths generated by evolved and un-evolved FLC1, we can find that the path length was decreased at 7.84%, while the running time was decreased at 13.80%.

Otherwise, the results of the navigation of multiple mobile robots in the environments with 2 static and 4 dynamic obstacles are shown in Fig. 8. In this situation, all the robots use the same navigation strategies and the same optimized FLC1 identically. Compare the paths generated by evolved and un-evolved FLC1, we can find that the path length of three robots was respectively decreased at -2.47%, 17.02%, 36.06%, and the running time was decreased at 5.00%, 34.13%, 47.95%.

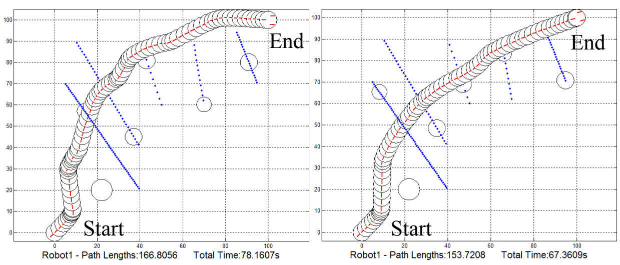

Fig. 7 The path of one robot before (left) and after (right) evolution.

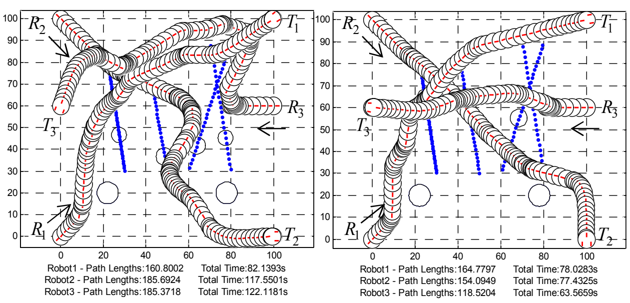

Fig. 8 The path of multiple robots before (left) and after (right) evolution.

6. Conclusions

This paper studied a navigation problem for multiple mobile robots in unknown dynamic environment using optimized fuzzy logic controller. The information of environment and the distance between the robot body and surrounding obstacles were detected all by the robot's own sensors. The genetic algorithm was used to optimize the membership functions and rules set of FLC. The simulation results showed that the FLCs have excellent performance in path planning and obstacle avoidance process. Compare the paths generated by the optimized and un-optimized fuzzy controller the running time of the navigation procedure was observably decreased.

In this paper, the dynamic obstacles move under preset trajectories, to perform the simulation in the environment with randomly moving obstacles will be a rewarding challenge for the future work. In addition, all the obstacles in this work are simulated as roundness, thus, in the future we will extend our experiment to the environment with dynamic irregular obstacles.

References

[1] D. Janglová, "Neural Networks in Mobile Robot Motion," International Journal of Advanced Robotic Systems, Vol. 1, No. 1, pp. 15-22, 2004.

[2] C. C. Hsu, Y. C. Liu, "Path planning for robot navigation based on Cooperative Genetic Optimization," Networking, Sensing and Control (ICNSC), 2014 IEEE 11th International Conference on, IEEE, pp. 316-321, April 2014.

[3] R. Carelli, C. M. Soria and B. Morales, "Vision-based Tracking Control for Mobile Robots," in Proceedings of 12th International Conference on Advanced Robotics, pp. 148-152, July 2005.

[4] P. K. Padhy, T. Sasaki, S. Nakamura and H. Hashimoto, "Modeling and Position Control of Mobile Robot," in Proceedings of 11th IEEE international Workshop on Advanced Motion Control, pp. 100-105, Mar. 2010.

[5] S. S. Ge, Y. J. Cui, "Dynamic motion planning for mobile robots using potential field method," Autonomous Robots, Vol. 13, No. 3, pp. 207-222, 2002.

[6] Van Den Berg J. P., Overmars, M. H., "Roadmap-based motion planning in dynamic environments," Robotics, IEEE Transactions on, Vol. 21, No. 5, pp. 885-897, 2005.

[7] C. G. Zhang, Y. G. Xi, "Rolling path planning and safety analysis of mobile robot in dynamic uncertain environment," Control Theory & Applications, Vol. 20,

No. 1, pp. 37-44, 2003.
[8] M. Wang, James N.K. Liu, “Fuzzy Logic Based Robot Path Planning in Unknown Environment,” in Proceedings of 4th International Conference on Machine Learning and Cybernetics, vol. 2, pp. 813-818, Guangzhou, China, Aug. 2005.
[9] H. M. Alfaro, S. G. Garcia. “Mobile Robot Path Planning and Tracking using Simulated Annealing and Fuzzy Logic Control,” Expert Systems with Applications, vol. 15, pp. 421-429, 1998.
[10] D. Zhao, T. Zou, “A finite-time approach to formation control of multiple mobile robots with terminal sliding mode,” International Journal of Systems Science, vol. 43, no. 11, pp. 1998-2014, 2012.
[11] X. Y. Zhong, X. G. Zhong, X. F. Peng, “Velocity-Change-Space-based dynamic motion planning for mobile robots navigation,” Neurocomputing, vol. 143, no. 2, pp. 153-163, Nov. 2014.
[12] H. K. Lee, D. H. Lee, R. Zhao, G. K. Lee, M. Lee, “On Parameter Selection for Reducing Parameter Convergence of Genetic Algorithm”, 23rd International Conference on Computer and Their Applications in Industry and Engineering (CAINE-2010) pp. 214-219, Las Vagas, USA, 2010.

저 자 소 개

조 연 (Ran Zhao)

He received his B.S. degrees in Department of Information Engineering from Shandong University, China, in 2006, and the M.S. degree in Department of Electrical and Electronics Engineering from Korea University of Technology and Education, Korea, in 2009. He is currently working toward a Ph.D. degree. His research interests include intelligent mobile robot, fuzzy logic control and genetic algorithm.

이 홍 규 (Hong Kyu Lee)

He received his B.S., M.S, and Ph.D. degrees in Department of Electronics Engineering from Seoul National University, Seoul, Korea in 1977, 1979, and 1989, respectively. He was with the Agency of Defense Development from 1979 to 1992. Since 1992, Department of Electrical Engineering, Korea University of Technology and Education, where he is currently a Professor. His major research interests include control theory and intelligent mobile robot, fuzzy logic control, neural networks, and genetic algorithms.