A KNOWLEDGE BASE FOR DYNAMIC PATH PLANNING OF MULTI-AGENTS

Nosan Kwak **robot97@snu.ac.kr**

Sanghoon Ji **robot91@snu.ac.kr**

Beomhee Lee **bhlee@asri.snu.ac.kr**

School of Electrical Engineering
Seoul National University
Seoul, Korea

Abstract: A fuzzy rule base is proposed to navigate multi-agents from initial positions to target positions in unknown environments. The proposed fuzzy rule base determines the highest priority of nine possible heading directions. The fuzzy rule base has been developed employing genetic algorithms as an approach to dynamic path planning of autonomous multi-agents in unknown environments. Paths which satisfy some optimization criteria with respect to moving distance, smoothness, and clearance of obstacles was obtained from the fuzzy rule base. The fuzzy rule base was obtained from off-line navigation with precise sensor modeling and applied to various simulated on-line navigation. The performance of the fuzzy rule base in different unknown environments is acceptable and shown in simulation results.

Keywords: Fuzzy control, Genetic algorithms, Path planning, Mobile robots.

1. INTRODUCTION

The mobile robot path planning problem is typically formulated as follows: given a robot and a description of an environment, plan a path between two specified locations which is collision-free and satisfies certain optimization criteria (Xiao, *et al.*, 1994). In multi-agent cases, other robots are considered as moving obstacles when there is no communication among robots. Path planning within moving obstacles is so complicated that a robot should predict other robots' motions (Pratihar, *et al.*, 1999). In this case, it is not efficient to design paths themselves because an environment changes. Therefore, it is more attractive to design a knowledge base for path planning of multiple robots. Another problem in path planning is real-time navigation. Traditional off-line planners often assume that an environment is perfectly known and try to search for the optimal path based on some fixed criteria which are usually costly (Latombe, 1991; Yap, 1987). On-line planners, on the other hand, are often purely reactive and do not try to optimize a path (Arkin, 1989; Brooks, 1986).

Over the years, researchers have found suitability to plan paths in using techniques such as Neural Networks (NNs) (Yang and Luo, 2004), Genetic Algorithms (GAs) (Xiao, *et al.*, 1994), Fuzzy Logics (FLs), and hybrid methods (Juidette and Youlal, 2000; Pratihar, *et al.*, 1999). These methods have capabilities of handling imprecision and uncertainty with a reasonable amount of computational complexity. And domain-specific knowledge is used for path planning (Yu and Yang, 2004). There are a lot of works related to path planning of a single robot comparing to that of multi-agents. Path planning of multi-agents in an environment with obstacles was solved using graph optimization algorithms (Kolushev and Bogdanov, 2000). CAPD which performs path planning via checkpoint and dynamic priority assignment, using statistical estimates of the environment's motion structure was proposed (Olive, *et al.*, 2000). A Genetic-Fuzzy approach to path planning of multi-agents provides general, flexible, low computation complexity, and adaptive planners. The use of fuzzy logic techniques help in quickly determining imprecise yet obstacle-free paths, and the use of genetic algorithms help in learning a near optimal rule set that a robot should use while navigating in present of moving obstacles (Pratihar, *et al*., 1999). A fuzzy rule base for path planning of a single robot is developed using genetic algorithms (Juidette and Youlal, 2000).

In this paper, a fuzzy rule base is developed for path planning of three robots when their initial and target positions are known. A rule in the fuzzy base consists of infrared sensor inputs, a direction from a robot's direction to a target, and a priority of a heading direction. Only the fuzzy rules are evolved by genetic algorithms, and fuzzy membership functions are designed by an expert. For real-time autonomous navigation, each robot should be capable of sensing its environment, planning a real-time route from the initial position to the target position without collision, and controlling direction and velocity. Many researchers developed path planners simplifying sensor modeling, which they were not adequate to apply to real environments (Berman, Edan, and Jamshidi, 2003). It is presented that infrared proximity sensors are modeled precisely based on the real sensor specification. After the fuzzy rule base is developed using genetic algorithms in off-line navigation, it is applied to simulated on-line navigation. On-line navigation employing the obtained fuzzy rule base in various environments was successful. The results are shown in simulations.

2. DESIGN OF A FUZZY CONTROLLER

2.1 Problem Statement

The problems in this work are formulated as follows: How do three robots reach each target position of them when an environment is unknown? How can good paths which satisfy the shortest moving distance, smoothness, and clearance of obstacles be obtained? To solve these problems, a fuzzy rule base is proposed in this work. It is assumed that 1) errorless localizations of robots, 2) initial and target position of each robot are known, 3) each robot doesn't know other robots' locations, and 4) a circular type robot has only infrared proximity sensors. A fuzzy controller as a knowledge base for path planning of multi-agents makes it helpful for robots to navigate in totally unknown environments. The objective of this paper is to develop the fuzzy controller to make multi-agents reach their target positions with the optimization criteria. Infrared proximity sensors and a direction from a robot's direction to a target position are used as inputs to the fuzzy controller. Thus, it is required to model the sensor precisely in order to apply the fuzzy controller to on-line navigation. Each robot shares one fuzzy rule base, thus only one fuzzy rule base is required regardless of the number of robots. However, the more the number of robots is, the more complexity increases. Thus, if the number of robots increases it will be more difficult to find a proper fuzzy rule base, and it will take much longer time to obtain the fuzzy rule base. In this paper, the variation of the number of robots is not considered.

2.2 Modeling of a Mobile Robot

Realistic sensor modeling is a crucial matter when on-line navigation is performed, because infrared sensors have errors and own characteristics: narrow sensing range, specular reflection. Each robot has 9 infrared proximity sensors which cover up 45° of frontal areas.

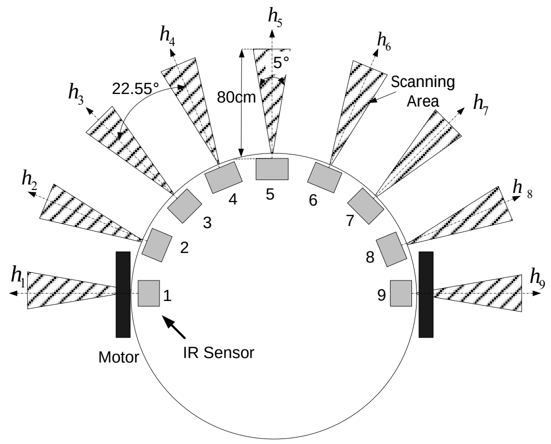


Fig. 1. Mobile robot model and definition of heading directions h_i, i={1,2,…,9}.

In Figure 1, the model of a mobile robot with infrared proximity sensors is shown. The output of the sensor is the distance from the sensor to an object. The sensing range is up to 80cm with a 5° aperture angle. Corresponding to the frontal areas, nine possible heading directions are defined in Figure 1.

2.3 The Fuzzy Controller for Path Planning

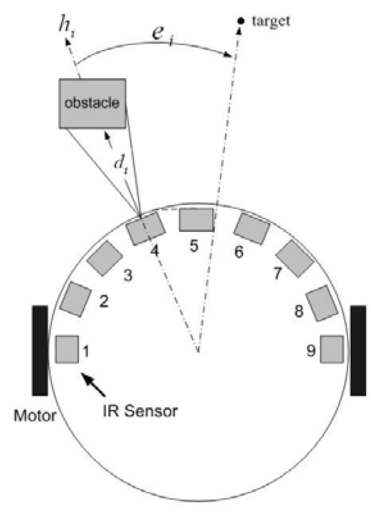


Fig. 2. Fuzzy linguistic variables d_i which is the distance of the obstacle measured by ith sensor and e_i which is the angle from h_i to a target.

Performance of a fuzzy controller highly depends on a choice of a fuzzy rule base. The proposed fuzzy controller has two input linguistic variables d_i which is the distance of the obstacle measured by *i*th sensor and e_i which is the angle from h_i to a target. The input linguistic variables are shown in Figure 2. The output linguistic variable p_i is the priority of the input variable set (d_i, e_i), i={1,…,9}. Each input linguistic variable has five linguistic values. The linguistic variable e_i has five linguistic values because the orientation from a robot to its target position is very sensitive to make the robot reach its target position. Therefore, there are totally 25 fuzzy rules, which comprise a fuzzy rule base. The linguistic values and their meanings are presented in Table 1.

Table 1 Linguistic values and their meanings

d_i (Distance)		e_i (Orientation)	
VN	Very Near	VR	Very Right
N	Near	R	Right
M	Medium	F	Front
F	Far	L	Left
VF	Very Far	VL	Very Left

An example of a fuzzy rule base is as follows:

$$rule\ 1: IF\ d_i = VN\ and\ e_i = VR\ THEN\ p_i = \mathrm{C}_1$$
$$rule\ 2: IF\ d_i = N\ and\ e_i = VR\ THEN\ p_i = \mathrm{C}_2$$
$$\vdots$$
$$rule\ 25: IF\ d_i = VF\ and\ e_i = VL\ THEN\ p_i = \mathrm{C}_{10}$$

where $C_1, C_2, \ldots, C_{10}$ are linguistic values of p_i that satisfy $1 \le C_j \le 10$, $j=\{1,2,\ldots,10\}$, and the number 10 denotes the highest priority. The priority of each heading direction is obtained by the 25 fuzzy rules. Thus, the number of fuzzy rules is not dependent on the number of heading directions but on the input linguistic variables and values. The d_i, $i=\{1,\ldots,9\}$, is defined in the universe of discourse U_d= [0, 80] to represent the measured distance by *i*th sensor. A fuzzy set D_i on d_i is defined as follows:

$$D_i = \int_{U_d} \mu_{D_i}(d_i)/d_i \qquad (1)$$

In the same way, the e_i is defined in the universe of discourse U_e= $[-\pi/2, \pi/2]$. Fuzzy sets E_i on e_i is defined as follows:

$$E_i = \int_{U_e} \mu_{E_i}(e_i)/e_i \qquad (2)$$

Likewise, the output linguistic variable p_i is defined in the universe of discourse U_p= [1, 10]. Fuzzy sets P_i on p_i is defined as follows:

$$P_i = \int_{U_p} \mu_{P_i}(p_i)/p_i \qquad (3)$$

For assigning the priority of each input variable set (d_i, e_i), i={1,…,9}, the center of area method as a defuzzification is used. The membership functions of inputs and an output are all triangular forms, which are shown in Figure 3. The design of fuzzy sets is done by an expert.

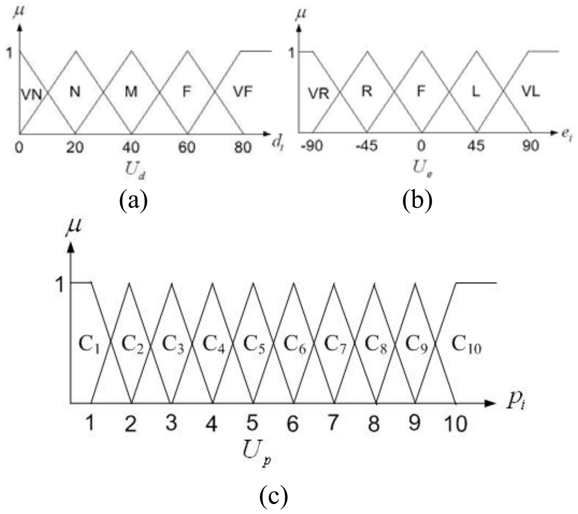


Fig. 3 Membership functions of measured distance d_i (a), orientation from a heading direction to a target (b), and the priority of each heading direction (c)

The fuzzy rule base R^i for *i*th heading direction can be represented as union as follows:

$$R^i = \left\{ \bigcup_{n=1}^{N} R_n \right\} = \left\{ \bigcup_{n=1}^{N} (D_i \times E_i) \to P_i \right\} \qquad (4)$$

where N is the number of rules. The highest priority among all input sets (d_i, e_i), i={1,…,9} can be obtained after all priorities of heading directions are computed by the fuzzy rule base. But for the robustness of the rule base, especially for avoiding collisions, a heading direction should consider its adjacent priorities. The state of a heading direction is evaluated by the following equation:

$$state\_h_i = \sum_{j=1}^{5} W_j \bullet p_{i+j-3}, \quad i = 1,2,\ldots,9 \qquad (5)$$

Where $\boldsymbol{W}$ is a weight coefficient vector which is predefined as $\boldsymbol{W}$={0.5, 0.7, 1.0, 0.7, 0.5}, and W_j is the *j*th element of $\boldsymbol{W}$. The heading direction which has the largest state value is selected for the next robot's heading direction. The linear velocity of the robot is scaled by the sparseness of the environment in a time step.

3. Evolution of the Fuzzy Rule Base

A genetic-fuzzy approach is proposed to create the proper fuzzy rule base. Genetic algorithms are used here to refine fuzzy rule bases which are randomly created. Steady-state genetic algorithms are used because they usually make chromosomes converge fast. Also, elitism is used to protect the best fuzzy rule base from removing in a population. In this section, the genetic algorithms and the way how to apply the knowledge base resulted from the genetic operations to simulated on-line navigation are described. Simulated off-line navigation and on-line navigation are depicted in Figure 4.

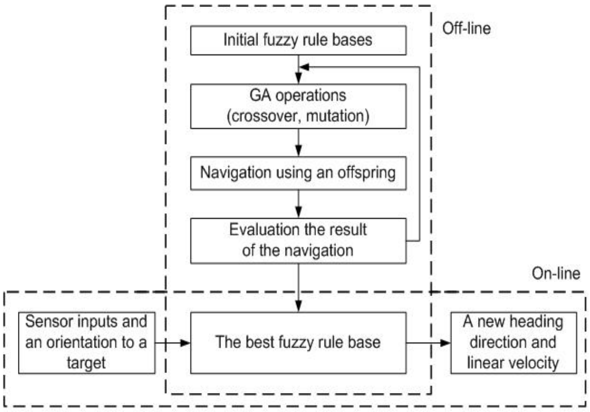


Fig. 4 Proposed off-line and on-line processes

3.1 Design of a Chromosome

Representation of a chromosome is a key issue in the work of GAs. In this work, only fuzzy rules are encoded in a chromosome. There are two inputs which have five linguistic values for each. Thus, the total number of rules is 25, and there is an output which has ten linguistic values, thus the total number of consequents is 10. All input cases are considered to get a precise fuzzy controller which can handle complex situations. One chromosome represents one fuzzy rule base. All robots share one fuzzy rule base, thus only one fuzzy rule base is needed for three robots. The consequent in this work represents the priority of an input set. All consequents are assigned to an integer from 1 to 10. An example of a chromosome is shown in Figure 5.

index:	1	2	3	4	5	6	7	…	25
	2	3	1	7	1	10	8	…	5

Fig. 5 An example of a chromosome

The site of a gene represents inputs of a fuzzy rule, and the value in the gene represents a priority. For instance, the fuzzy rule base in Figure 5 is like this:

$$rule\ 1: IF(d_i = VN)\ and\ (e_i = VR)\ \ THEN\ p_i = 2$$
$$rule\ 2: IF(d_i = N)\ and\ (e_i = VR)\ \ THEN\ p_i = 3$$
$$rule\ 3: IF(d_i = M)\ and\ (e_i = VR)\ \ THEN\ p_i = 1$$
$$\vdots$$
$$rule\ 25: IF(d_i = VF)\ and\ (e_i = VL)\ \ THEN\ p_i = 5$$

3.2 Genetic Operators

Chromosomes are selected by roulette wheel selection in a population. Two selected chromosomes are processed by three genetic operators: crossover, mutation, and elitism. The one point crossover operator is used. The mutation operator selects a random site and changes its value to the nearest value. In case a robot falls into a local minimum while navigating an environment, the control system makes the robot avoid the stuck situation by changing the robot's heading direction. It is easy to know when collisions occur, but it is difficult when local minima occur. To know when local minima occur, a simple technique, which is shown in Figure 6, is proposed. In Figure 6, the value of each cell represents how many times it is occupied. For example, '0' means never occupied and '15' means occupied 15 times. If a robot falls into a local minimum, it will occupy the same position many times. If the number of occupation of a place is bigger than a threshold which is decided by lots of experiments, then it is a local minimum case.

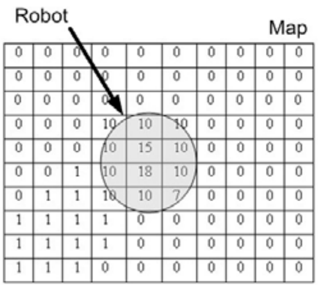


Fig. 6 Technique for detecting a local minimum

The elitism operator prevents the best chromosome in the previous generation from removing in the next generation. It generally makes chromosomes in a population converge quickly.

3.3 Evaluation of a Chromosome

Every chromosome doesn't make all robots go to their target positions without collisions. Some chromosomes make robots collide with walls or other robots. Thus feasible paths which make robots reach their target positions with collision-free and infeasible paths which make collisions or unreachable situations should be evaluated differently. If paths are feasible each robot saves its position in each step, and the path of the robot is formed by line segments connecting positions. The evaluation function for a feasible path $eval_f$ is designed to accommodate three different optimization goals: 1) minimize distance travelled (*dist*), 2) maintain a smooth path (*smooth*), and 3) satisfy the clearance requirements (the robot should not approach obstacles too closely) (*clear*) (Xiao, *et* *al.*, 1994). The evaluation function for a feasible path is as follows:

$$eval_f(T_i) = w_d \cdot dist(T_i) + w_s \cdot smooth(T_i) + w_c \cdot clear(T_i) \quad (6)$$

where the constants, w_d, w_s, w_c represent the weights on the total cost of the path's length, smooth, and clearance, respectively, and T_i is the path of *i*th robot, $i=\{1,2,3\}$. The evaluation function of an infeasible path $eval_{inf}$ is designed to accommodate two criteria: 1) minimize the number of collisions (*nCollisions*) and 2) minimize the number of travelling steps (*nSteps*).

$$eval_{inf}(T_i) = w_{col} \cdot nCollisions(T_i) + w_{step} \cdot nSteps(T_i) \quad (7)$$

where the constants, w_{col}, w_{step} represent the weights on the cost of the path's collisions and travelling steps, respectively. The w_{col} is determined by lots of experiments and has quite bigger value in comparison with other constants because chromosomes which make collisions are the worst. And there is a step-limit for some chromosomes which don't make robots reach their target positions. The smaller value of fitness is, the better the chromosome is. The objective is to minimize fitness of both feasible and infeasible paths. The fitness of a chromosome is the average of all paths' evaluations. The evaluation of an infeasible path is not always bigger than that of a feasible path. This is because a chromosome which makes infeasible paths might become a good chromosome which makes feasible paths after certain genetic transformations. Besides, this different evaluation strategy allows some overlap between fitness of feasible and infeasible paths because a very poor feasible path is not necessarily better than a very good near-feasible paths in the sense of evolving solutions.

3.4 On-line Navigation

The best fuzzy rule base resulted from GA operations can be applied to simulated on-line navigation. All robots should know their initial and target positions. The fuzzy controller assumes that the localization of each robot is error-free. Thus, the localization problem should be considered when multi-agents navigate in real environments. In this paper, only simulated navigation is treated.

4. SIMULATION RESULTS

A grid map whose size is 256 x 256 cm was used. The radius of a robot is 4.5cm, and the radius of a target position is 6cm. Obstacles were placed various positions in the map. A robot in both off-line and on-line navigation knows only where its target position is. Robots sense environments through infrared sensors modeled in Section 2.2.

4.1 Off-line Navigation and Results

Initial 100 chromosomes were randomly created and formed an initial population. The initial population was evolved for 100 generations. The crossover and mutation probabilities were set to 0.9, and 0.2, respectively. Because steady-state genetic algorithms were used, a produced offspring replaced the worst chromosome in a population. The map to evolve the fuzzy controller in off-line is described in Figure 7. Rectangular boxes represent obstacles. Also, there are three robots and their target positions which are presented by circles in the map. The experimental environment was made complex so that robots do a lot of works, which made all genes of a chromosome evolved. The objective of this paper is to make the robots reach their target positions with the suggested criteria and no collision. In Figure 7, 'R1' represents the first robot, and 'T1' represents the first robot's target position. In the same way, 'R2', 'R3', 'T2', and 'T3' represent the second robot, the third robot, the second robot's target position, and the third robot's target position, respectively. A target position is not a point but areas. Initial and target positions were added 10% noise to fixed positions.

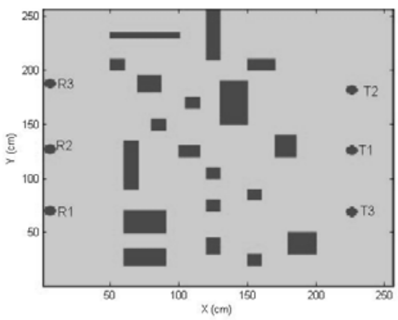


Fig. 7 Environment for evolving the fuzzy rule base

There were two bad cases when robots navigate: 1) colliding with walls, static obstacles, and other robots 2) not reaching their target positions. For the first case, (7) was employed to evaluate them. To solve the second case, the limit of the traveling steps was set, and the chromosomes were evaluated by (7). In early generations, the best fitness has radically decreased because the chromosome which makes feasible paths was generated. After about 30th generation, the best fitness has not decreased significantly. The best fuzzy rule base is presented in Table 2.

Table 2 The best fuzzy rule base by GAs

	VN	N	M	F	VF
VR	4	3	9	5	8
R	8	1	5	7	9
F	1	4	5	6	9
L	3	3	7	7	9
VL	2	3	4	5	4

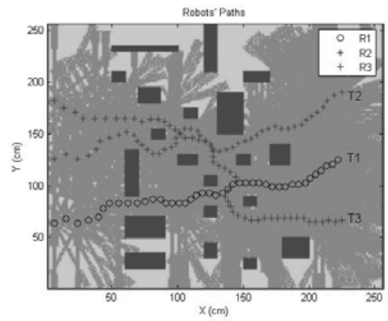
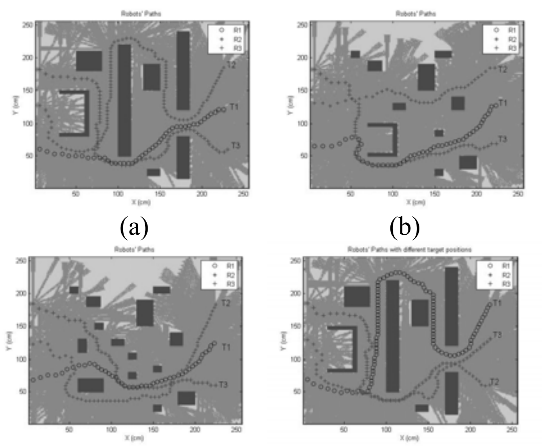


Fig. 8 Robots' paths by the best chromosome

The average of the initial fitness was about 6169. Coefficients, w_d, w_s, w_c, w_{col}, w_{step} in the evaluation functions are set to 1, 10, 5, 500, 10, respectively. Robots' paths by the best chromosome are shown in Figure 8. The 'R1', 'R2', and 'R3' have reached their target positions with 41, 47, and 51 steps, respectively. The navigation took 8 seconds, and the fitness was 629.74. The dark parts in the environment present sensed areas by sensors.

4.2 On-line Navigation and Results

With the best chromosome resulted by off-line navigation, simulated on-line navigation was performed in many different unknown environments. All coefficients employed in off-line navigation were used in on-line navigation. All cases in Figure 9 were successful navigation without collisions. In Figure 9(d), though the target positions were different than others, the navigation was successful.



(a) (b) (c) (d)

Fig. 9 Various on-line navigation results. The target positions in (d) are different from (a), (b), and (c).

Summary of navigation results is shown in Table 3.

Table 3 Results of on-line navigation

	Total steps	Fitness	Running time	Collision
(a)	214	759.62	12.781 s	No
(b)	147	651.04	8.719 s	No
(c)	166	675.76	13.156 s	No
(d)	224	775.81	17.625 s	No

In Table 3,'Total steps' means the sum of the number of steps of each robot moved, and 'Running time' means navigation time in simulations measured by a computer which has 2.4GHz CPU clock and 512MB RAM. All on-line experiments were collision-free. When robots navigated the environment in Figure 9(c) for 100 times, the number of navigation without collision was 76. This is because of the added noise to the initial and target positions.

4. CONCLUSION

In this paper, the fuzzy controller as a knowledge base for dynamic path planning of three robots was developed with precise sensor modeling. Robots have obtained the proper fuzzy rule base for several unknown environments. But, the fuzzy rule base caused collisions in some environments though the rate of successful navigation was high. It costs high to obtain the proper fuzzy rule base by off-line navigation, but once the proper fuzzy rule base obtains, there is no severe computational cost to navigate in on-line. In case the fuzzy rule base is applied to real environmental navigation with localizations of robots, all robots can reach their target positions which are known. As further research, the control system should have learning abilities in on-line navigation. The learning abilities can save time to obtain the better fuzzy rule base.

ACKNOWLEDGMENT

This research was performed for the Intelligent Robotics Development Program which is one of the 21st Century Frontier R&D Programs funded by the Ministry of Science and Technology of Korea and the HWRS-ERC at KAIST

REFERENCES

Fukuda, T, and N. Kubota (1999) An intelligent robotic system based on a fuzzy approach. *Proceedings of the IEEE*, vol. 87, no. 9.

Juidette, H., and H. Youlal (2000) Fuzzy dynamic path planning using genetic algorithms. *Electronics Letters*, vol. 36, no. 4, pp. 374-376.

Lee, T.L., and C.J. Wu (2003) Fuzzy motion planning of mobile robots in unknown Environments. *IJournal of Intelligent and Robotic Systems 37,* pp. 177-191.

Pratihar D.K., K. Deb, and A. Ghosh (1999) A genetic-fuzzy approach for mobile robot navigation among moving obstacles. *International Journal of Approximate Reasoning 20*, pp. 145-172.

Xiao, J., Z. Michalewicz, L. Zhang, and K. Trjoanowski (1994) Adaptive Evolutionary Planner/Navigator for Mobile Robots. *IEEE Trans. Evolutionary Computation*, vol. 1, no 1, pp. 18-28.