A Sensor-Based Navigation for a Mobile Robot Using Fuzzy Logic and Reinforcement Learning

Hee Rak Beom and Hyung Suck Cho

***Abstract*— This paper proposes a sensor-based navigation method which utilizes fuzzy logic and reinforcement learning for navigation of mobile robot in uncertain environments. The proposed navigator consists of an avoidance behavior and goal-seeking behavior. Two behaviors are independently designed at design stage and then combined by a behavior selector at running stage. A behavior selector using a bistable switching function chooses a behavior at each action step so that the mobile robot can go for the goal position without colliding with obstacles. Fuzzy logic maps the input fuzzy sets representing the mobile robot's state space determined by sensor readings to the output fuzzy sets representing mobile robot's action space. Fuzzy rule bases are built through the reinforcement learning which requires simple evaluation data rather than thousands of input-output training data. Since fuzzy rules for each behavior are learned through reinforcement learning method, fuzzy rule bases can be easily constructed for more complex environments. In order to find mobile robot's present state, the ultrasonic sensors mounted at the mobile robot are used. The effectiveness of the proposed method is verified by a series of simulations.**

I. Introduction

PATH planning is one of the most vital task in navigation of autonomous mobile robots. Path planning for mobile robot may be divided into two categories: One is a global path planning based on a priori complete information about the environment and the other is a local path planning based on sensory information in uncertain environment where the size, shape and location of obstacles are unknown. The global path planning method includes configuration space method [1], potential field method [2], generalized Voronoi diagram [3], and graph search method [4]. These methods have been carried out in off-line manner in completely known environments. However, these methods are not suitable for navigation in complex and dynamically changing environments where unknown obstacles may be located on a priori planned path. Thus, the sensor-based local path planning, so called obstacle avoidance, carried out in on-line manner is required in the navigation of mobile robots.

Local path planning utilizes the information provided by sensors such as ultrasonic sensor, vision, laser range finder, proximity sensor and bumper switch. Brooks [5] applied the force-field concept to the obstacle avoidance problem for mobile robots equipped with ultrasonic sensors whose readings are used to compute the repulsive forces. Borenstein and Koren [6] proposed the virtual force field method for fast running mobile robots equipped with ultrasonic sensors and addressed the inherent limitations of this method [7]. The above methods have shortcoming, namely, it is difficult to find the force coefficients influencing on the velocity and direction of mobile robots in cluttered environments which can not be described as a mathematical model. In order to overcome the above problem, fuzzy logic and neural network approaches have been employed in navigation of mobile robot.

Fuzzy logic approach has an advantage that it deals with various situations without analytical model of environments. Recently, many researchers proposed the navigation algorithms using the fuzzy logic. Takeuchi [8] proposed the hallway following method by extracting the features from simple hallway image. Ishikawa [9] presented a sensor-based navigation method using fuzzy control in an indoor environment and concentrated on constructing an efficient control knowledge base. However, in cases of operating the mobile robot in complex environments, the above methods have disadvantage that it is difficult to consistently construct rule bases because there are many situations to be considered for navigation. Also, it takes much time to tune the constructed rules because tuning operation depends on the expert's knowledge.

In order to overcome the above difficulties in constructing and tuning the rules, an error back propagation neural network using information received from external sensors has been recently used in navigation of mobile robot [10], [11]. Error back propagation neural networks require a sufficient set of representative patterns to characterize the cluttered environments so that the trained networks may act as a good navigator in the environments. However, it is not easy to obtain the training patterns which contain no contradictory input/output pairs. This is because the training data must be generated by moving the mobile robot in a representative pattern a few times in a supposedly similar way.

In the previous work [12], a navigation method using fuzzy logic and error back propagation neural network was proposed. The neural network, whose inputs are ultrasonic sensor readings, informs the mobile robot of the situation of environment in which mobile robot is at the present instant. Then, the mobile robot decides its next action using only a fuzzy rule base associated with the situation. This method can be implemented on a real time base because a small number of rules are employed to decide the next action. Furthermore,

Manuscript received January 8, 1993; revised April 27, 1994.

H. R. Beom is with the Laboratory for Control Systems and Automation, Department of Precision Engineering and Mechatronics, Korea Advanced Institute of Science and Technology, Taejon, 305-701, Korea.

H. S. Cho is with the Center for Robotics and Automation, Department of Precision Engineering and Mechatronics, Korea Advanced Institute of Science and Technology, Taejon, 305-701, Korea.

IEEE Log Number 9405073.

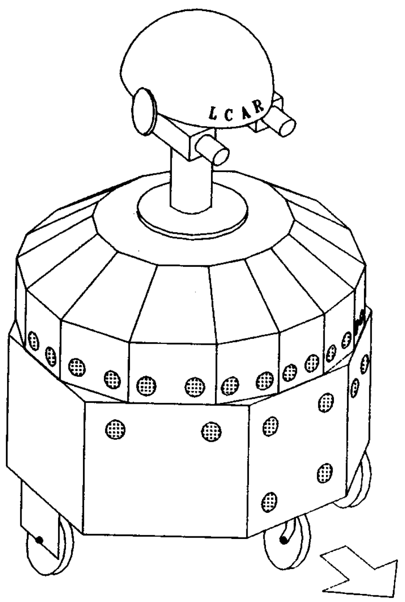


Fig. 1. The mobile robot, LCAR.

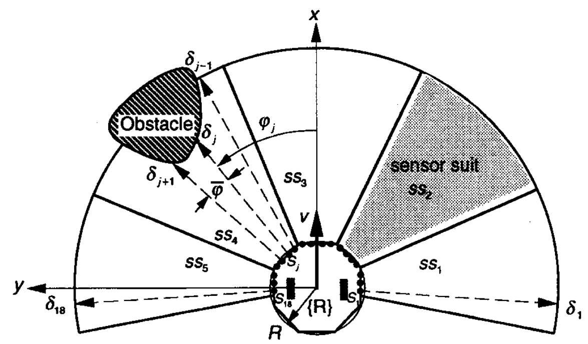


Fig. 2. The arrangement of ultrasonic sensors and sensor suits.

contradictory rules can be easily excluded from rule base since one rule base is responsible for one situation. The disadvantage of this method however is that constructing and tuning the rule bases depend on the expert's knowledge.

To avoid such a criticism, a new navigation method using fuzzy logic and reinforcement learning is proposed in this paper. Whenever a mobile robot navigates in uncertain environment towards the goal position, avoidance behavior and goal-seeking behavior always conflict. The avoidance behavior is used to avoid the obstacles irrespective of the goal position, while the goal-seeking behavior is used to seek the goal position irrespective of obstacle location. It is necessary for a navigator to efficiently combine two behaviors. For this, two behaviors are independently designed by fuzzy logic and reinforcement learning and are combined by the action of a switching function according to situations around the mobile robot. The fuzzy logic is used to represent the mapping between the sensor input space and mobile robot action space. The correct mapping is found by reinforcement learning [13], [14]. Fuzzy rule bases are built by input and output fuzzy sets which quantize the sensor input space and the mobile robot action space, respectively. The mobile robot action space consists of steering angle and linear velocity of mobile robot. Correct mapping from input state space to output action space is gradually accomplished by the reinforcement learning method as the mobile robot equipped with external sensors navigates in uncertain environments.

Reinforcement learning is performed based on a scalar evaluation value with which a supervisor provides learning system. A supervisor plays a role of providing the learning system with an evaluation value of the performance of a behavior according to a performance measure. Therefore, this learning method differs from the supervised learning method which requires the desired outputs for each given input. This learning method is suitable for the obstacle avoidance problem where it is difficult to determine the desired outputs for each situation. This navigation method is simulated for a wheeled mobile robot named as LCAR which has been developed in Laboratory for Control Systems & Automation, KAIST. As shown in Fig. 1, it is equipped with twenty six ultrasonic sensors, stereo cameras and structured light, and sensors for dead reckoning. The ultrasonic sensors are mounted on the front face of the mobile robot in three layers for the detection of obstacles with various heights. The cameras are fixed at the pan-tilt device mounted on the top of the mobile robot. The measurement range of ultrasonic sensors can be adjusted within 10m by the sensor control board developed in the laboratory. A series of simulations are carried out for various environmental conditions to investigate the validity of the proposed method. The simulation results show that the proposed navigator has the abilities to adapt to unknown environments by learning and thus, to navigate through the environments where a local minimum occurs.

II. ACQUISITION OF ENVIRONMENTAL INFORMATION

Mobile robot navigation encounters with static and dynamic environments. The dynamic environment can be defined as: any unregistered obstacles which move or change their shapes. In this study, eighteen ultrasonic sensors in the upper layer are used to understand environmental situations, for example, what types of obstacles exist in front of mobile robot or whether mobile robot can go through without colliding with obstacles. These sensors act as both a transmitter and a receiver. Fig. 2 shows the arrangement of the ultrasonic sensors and the sensors are marked as dots in the figure. The distances e_j, $(j = 1, 2, \cdots, 18)$, from the center of the robot frame $\{R\}$ to obstacles detected by jth sensor, S_j can be defined as $e_j = \delta_j + R$. Here, R is the radius of the robot and the δ_j is the jth sensor reading drawn by the dashed line. In this figure, φ_j denotes the orientation of the jth sensor with respect to the frame $\{R\}$. Thus, the orientations of ultrasonic sensors can be expressed by the relationship, $\varphi_j = (j - 9.5)\overline{\varphi}, j = 1, 2, \cdots, 18$. Here, the angle $\overline{\varphi}$ denotes the angle between two adjacent sensors and is fixed at $11.25°$. The number of 9.5 is required to express orientation angles of sensors installed at right and left hand sides from the x-axis by negative and positive values, respectively. The coordinate values of the

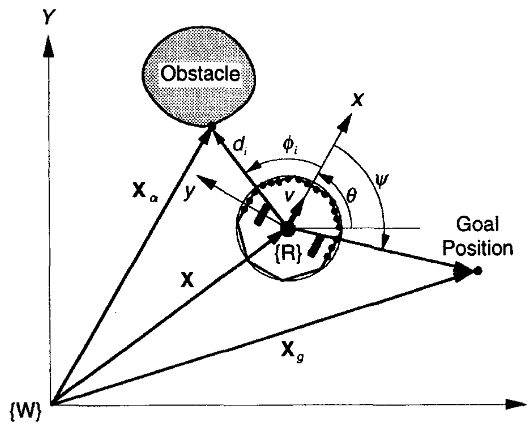


Fig. 3. The coordinate frames and control variables.

obstacle (x_{0j}, y_{0j}) detected by jth sensor with respect to the robot frame $\{R\}$ are expressed as

$$\begin{aligned} x_{0j} &= (\delta_j + R)\sin(\varphi_j) \\ y_{0j} &= (\delta_j + R)\cos(\varphi_j). \end{aligned} \tag{1}$$

It is assumed that the x-axis of the coordinate frame $\{R\}$ is aligned with the moving direction of mobile robot. As shown in Fig. 2, in order to avoid the increase of the dimension of input space, sensor suits are introduced. A sensor suit groups with some neighboring sensors. The first and fifth sensor suits, ss_1 and ss_5, are composed of 3 neighboring sensors, while other sensor suits are composed of 4 neighboring sensors. As shown in Fig. 3, the distance, d_i, measured by ith sensor suit from the center of mobile robot to obstacle and direction, ϕ_i of obstacle with respect to frame $\{R\}$ are defined, respectively, as

$$d_i = \begin{cases} R + \min\{\delta_j | j = 4i-4, 4i-3, 4i-2, 4i-1\} \\ \quad \text{if} \quad i = 2,3,4 \\ R + \min\{\delta_j | j = 2^{i-1}, 2^{i-1}+1, 2^{i-1}+2\} \\ \quad \text{if}\, i = 1,5 \end{cases} \tag{2}$$

$$\phi_i = \frac{\pi}{4}(i-3) \tag{3}$$

The detection sensitivity of the ultrasonic sensor varies with the angle of the reflecting object to the line-of-sight of transducer. Thus, when the reading of a sensor corresponds to the maximum range, we can not differentiate whether the obstacles are really located on the line-of-sight of ultrasonic sensor. Such situation where the sensor represents the maximum range can be divided into two cases. One is the open space where obstacles are not located on the line-of-sight of sensor. The other is the case when obstacle is located on the line-of-sight of sensor with the inclination angle larger than half beam width of sensor. Therefore, in order to acquire the sensor data available for mobile robot navigation, preprocessing the sensor data is required. Among the range data incoming from sensors, meaningful range, δ_j is defined as $\delta_j = \min\{\delta_{\max}, \max(\delta_j, \delta_{\min})\}$. Here, $\delta_{\max}$ and $\delta_{\min}$ denote the maximum and minimum ranges of ultrasonic sensor, respectively. Their technical specifications and beam pattern may be found in Polaroid [15]. They are controlled

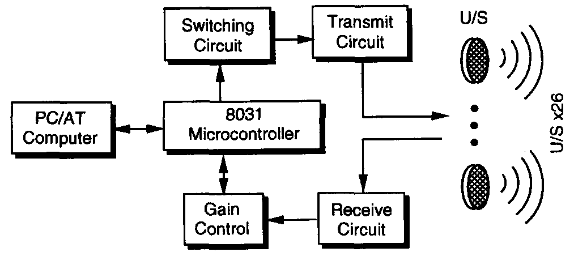


Fig. 4. The block diagram of the ultrasonic range finder.

by a microcontroller, Intel 8031, communicated with PC/AT computer. Their control boards consist of a transmitter module to fire the pulses, a receiver/gain-control module to receive the attenuated echo pulses and a switching module to arbitrarily select the activation of the sensor according to the sensing mode. The block diagram of the range finder is shown in Fig. 4. In the simulation study, it is assumed that the sensors can ideally detect the obstacles irrespective of inclination angle.

III. Navigation of Mobile Robot

Our goal is to design the navigator operating in uncertain environment. The proposed navigator consists of avoidance and goal-seeking behaviors. Fig. 5 shows a schematic diagram of the proposed navigator. Fuzzy decoder partitions the input space formed by the information acquired by sensors into some subspaces. The partitioned input space is represented by the rules. Each behavior is composed of fuzzy decoder, decision making, rule base, reinforcement learning mechanism and defuzzifier. Then, behavior selector by which the conflict between reaching the goal and avoiding the collision is resolved chooses the appropriate behavior based on the performance predictor. First, it is assumed that the action signals calculated by each behavior are applied to mobile robot controller before moving the mobile robot. Then, performance predictions of each behavior are evaluated by using the potential composed of repulsive and attractive potentials. The potential depends on the distance to the obstacles, linear velocity of mobile robot, the distance to the goal and the angular difference between mobile robot's moving direction and goal direction with respect to frame $\{R\}$. One of two behaviors is selected by the behavior selector operating based on the performance prediction value. The rule base of each behavior is constructed by fuzzy logic and each rule base is tuned by the reinforcement learning method. Design of the navigator composed of two behaviors proceeds in following sequences; (A) fuzzification of the input/output variables, (B) rule base construction through reinforcement learning, (C) reasoning process, (D) defuzzification of output variables and (E) behavior selection scheme.

A. Fuzzification of the Input-Output Variables

The goal of navigation in uncertain environments is to make the mobile robot arrive at destination point without colliding with obstacles. In order to accomplish this effectively, we must control the mobile robot motion in consideration of the

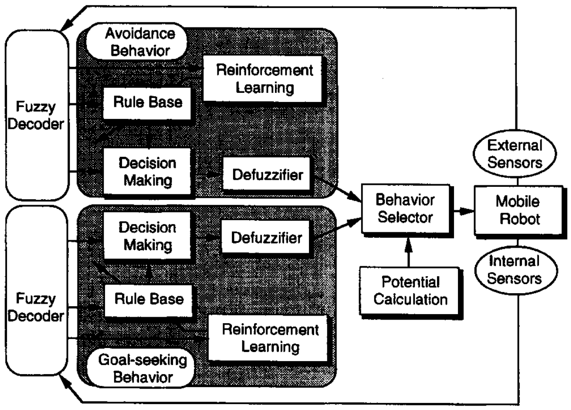


Fig. 5. The structure of the proposed navigator.

obstacle position (x_{0i}, y_{0i}), the mobile robot position (x, y) and its heading angle θ with respect to the world coordinate frame $\{W\}$ shown in Fig. 3. The motion of mobile robot can be realized by the control of its heading velocity, v and incremental steering angle, $\Delta\theta$. Thus, we choose the input variables for avoidance behavior as $d_i = \| \boldsymbol{X}_{0i} - \boldsymbol{X} \|$, $(i = 1, 2, \cdots, 5)$, and those for goal-seeking one as heading angle difference, ψ and distance to goal, $z = \| \boldsymbol{X}_g - \boldsymbol{X} \|$. The output variables for two behaviors are chosen as the incremental steering angle, $\Delta\theta$ and velocity, v. Here, the subscript, i denotes the number of the sensor suit for detecting the obstacles located in front of mobile robot and $\| \cdot \|$ denotes the Euclidean norm. The variable d_i is calculated by (2). The ψ is the angle between heading direction of the mobile robot and the direction of the goal position and the z is the distance from the current position, $\boldsymbol{X} = (x, y)$ to goal position, $\boldsymbol{X}_g = (x_g, y_g)$ with respect to the frame $\{W\}$. All the variables are defined in Fig. 3.

A fuzzy operator converts the crisp input data, z into the linguistic values, $\tilde{z}$ considered as labels of fuzzy sets and is defined as

$$\tilde{z} = \mathit{fuzzifier}(z) \tag{4}$$

where fuzzifier denotes a fuzzification operator. From now on, tilde sign ($\sim$) representing the fuzzy set will be omitted for simplicity. The input linguistic variables $d_i, (i = 1, 2, \cdots, 5), \psi$ and z are expressed by linguistic values (VN, NR, FR), (NB, NM, NS, ZZ, PS, PM, PB) and (VN, NR, FR, VF), respectively. The output linguistic variables v and $\Delta\theta$ are expressed by the linguistic values with membership functions having the triangular shaped functions shown in Fig. 6. Their center positions are going to be determined by reinforcement learning method. The linguistic terms have the following meanings:

VN:	*very near*	NR:	*near*
FR:	*far*	VF:	*very far*
NB:	*negative big*	NM:	*negative medium*
NS:	*negative small*	ZZ:	*zero*
PS:	*positive small*	PM:	*positive medium*
PB:	*positive big.*		

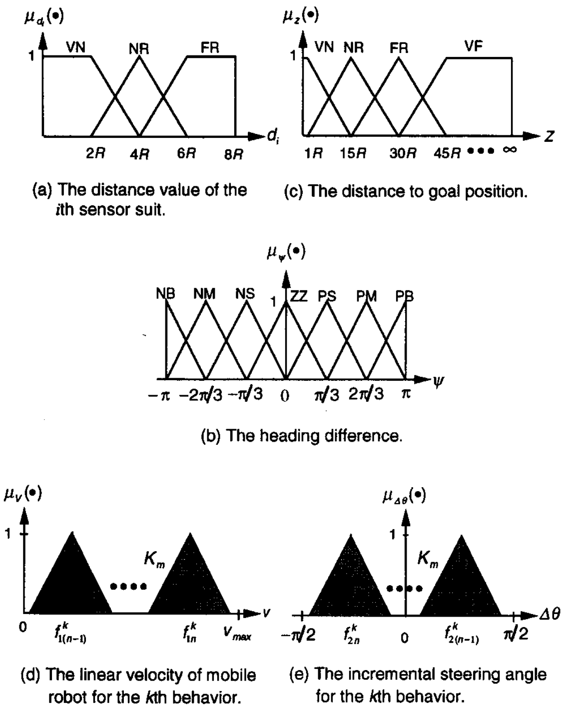


Fig. 6. The membership functions of the input-output variables. (a) The distance value of the ith sensor suit. (b) The heading difference. (c) The distance to goal position. (d) The linear velocity of mobile robot for the kth behavior. (e) The incremental steering angle for the kth behavior.

Fuzzy subsets contain elements with degree of membership, while fuzzy membership function $\mu_z(\cdot)$ of fuzzy set, z assigns a real number between 0 to 1 to every element in the universe of discourse. This membership value indicates the degree to which the element belongs to the fuzzy set. The membership functions of the fuzzy sets defined for two behaviors are shown in Fig. 6. Although fuzzy membership function can have various shapes depending on the designer's preference, in this study trapezoidal and triangular shapes are chosen to simplify the computation.

B. Rule Base Construction

The rule base for realizing each behavior can be constructed based on human experience. However, in the case of the complex environments, it is difficult to tune the rule base associated with each behavior by relying only on the human experience. In order to tune the rule bases effectively, the reinforcement learning method to be explained later is used. The rule bases for the behaviors consist of the rules taking the form of IF-THEN statements:

$$\begin{aligned}
R_1^1&\text{: IF } (d_1 = D_1) \text{ AND} \cdots (d_5 = D_1) \\
&\quad \text{THEN } (v = V_1, \Delta\theta = \Delta\Theta_1) \\
R_1^2&\text{: IF } (\psi = \Psi_1) \text{ AND } (z = Z_1) \\
&\quad \text{THEN } (v = V_1, \Delta\theta = \Delta\Theta_1) \\
&\quad \vdots
\end{aligned}$$

$$\begin{aligned}
R_n^1&: \text{IF}\,(d_1 = D_n)\,\text{AND}\cdots(d_5 = D_n)\\
&\quad \text{THEN}\,(v = V_n, \Delta\theta = \Delta\Theta_n)\\
R_n^2&: \text{IF}\,(\psi = \Psi_n)\,\text{AND}\,(z = Z_n)\\
&\quad \text{THEN}\,(v = V_n, \Delta\theta = \Delta\Theta_n)\\
&\vdots\\
R_{N_1}^1&: \text{IF}\,(d_1 = D_{N_1})\,\text{AND}\cdots(d_5 = D_{N_1})\\
&\quad \text{THEN}\,(v = V_{N_1}, \Delta\theta = \Delta\Theta_{N_1})\\
R_{N_2}^2&: \text{IF}\,(\psi = \Psi_{N_2})\,\text{AND}\,(z = Z_{N_2})\\
&\quad \text{THEN}\,(v = V_{N_2}, \Delta\theta = \Delta\Theta_{N_2})
\end{aligned}$$

where R_n^k, denotes the nth rules associated with the kth behavior. Also, Ψ_n, D_n, Z_n, V_n and $\Delta\Theta_n$ with $n = 1, 2, \cdots, N_k$ are the fuzzy linguistic values of linguistic variables $\psi, d_i, (i = 1, 2, \cdots, 5)$, and $\Delta\theta$ in the universe of discourse U_ψ, U_d, U_z, U_v and $U_{\Delta\theta}$, respectively. Here, N_k denotes the number of the rules associated with kth behavior. This nth rule R_n^k of the kth behavior can be represented as a fuzzy relation, R_n^1: $(D_n \times D_n \times D_n \times D_n \times D_n) \to (V_n, \Delta\Theta_n)$ for avoidance behavior and R_n^2: $(\Psi_n \times Z_n) \to (V_n, \Delta\Theta_n)$ for goal-seeking behavior. Here, $\to$ denotes fuzzy relation. The rule base for the avoidance behavior $(k = 1), R^k$ can be represented as the union

$$\begin{aligned}
R^k &= \left\{\bigcup_{n=1}^{N_k} R_n^k\right\}\\
&= \left\{\bigcup_{n=1}^{N_k} [(D_n \times D_n \times D_n \times D_n \times D_n) \to V_n,\right.\\
&\qquad \left.(D_n \times D_n \times D_n \times D_n \times D_n) \to \Delta\Theta_n]\right\}\\
&= \left\{\bigcup_{n=1}^{N_k} RV_n^k, \bigcup_{n=1}^{N_k} R\Theta_n^k\right\} = \{RV^k, R\Theta^k\} \qquad (5)
\end{aligned}$$

where R^k consists of the sub-rule-bases, RV^k and $R\Theta^k$, associated with control actions, v and $\Delta\theta$, respectively. Here, each sub-rule-base consists of N_k rules. The nth rules, RV_n^1 and $R\Theta_n^1$, for avoidance behavior $(k = 1)$ are the fuzzy relations in product spaces, $U_d \times U_d \times U_d \times U_d \times U_d \times U_v$ and $U_d \times U_d \times U_d \times U_d \times U_d \times U_{\Delta\theta}$, respectively. The nth rules, RV_n^2 and $R\Theta_n^2$, for goal-seeking behavior $(k = 2)$ are the fuzzy relations in product spaces, $U_\psi \times U_z \times U_v$ and $U_\psi \times U_z \times U_{\Delta\theta}$, respectively. Thus, the rules can be implemented as fuzzy relations with corresponding membership functions. The membership values of the nth rules, RV_n^1 and $R\Theta_n^1$, for avoidance behavior will be denoted by $\mu_{RV_n^1}(d_1, \cdots, d_5, v)$ and $\mu_{R\Theta_n^1}(d_1, \cdots, d_5, \Delta\theta)$, respectively. The membership values of the nth rules, RV_n^2 and $R\Theta_n^2$, for goal-seeking behavior will be denoted by $\mu_{RV_n^2}(\psi, z, v)$ and $\mu_{R\Theta_n^2}(\psi, z, \Delta\theta)$, respectively.

C. Reasoning Process

Utilizing the fuzzified input data, the fuzzy sets associated with the control actions of each behavior are determined by inferring from each rule base. When the inputs, $d_1 = d_1', \cdots, d_5 = d_5'$, are given, if they are fuzzy singletons, namely, $d_1' = D_1', \cdots, d_5' = D_5'$, the fuzzy control actions, V' and $\Delta\Theta'$, for the avoidance behavior are inferred from Zadeh's compositional rule of inference [16]. They are inferred by

$$V' = (D_1', D_2', D_3', D_4', D_5') \circ \bigcup_{n=1}^{N_1} RV_n^1 \qquad (6)$$

$$\Delta\Theta' = (D_1', D_2', D_3', D_4', D_5') \circ \bigcup_{n=1}^{N_1} R\Theta_n^1 \qquad (7)$$

where $\circ$ denotes the maximum-minimum composition. Here, $D_i', (i = 1, 2, \cdots, 5)$ are the fuzzy sets defined in the universe of discourse U_d. If Mamdani's minimum operation is applied to the fuzzy relations, the membership values of V' and $\Delta\Theta'$ are calculated by

$$\begin{aligned}
\mu_{V'}^1(v) = \bigcup_{n=1}^{N_1} \{&\mu_{D_n}(d_1') \wedge \mu_{D_n}(d_2') \wedge \mu_{D_n}(d_3')\\
&\wedge \mu_{D_n}(d_4') \wedge \mu_{D_n}(d_5') \wedge \mu_{V_n}(v)\} \qquad (8)
\end{aligned}$$

$$\begin{aligned}
\mu_{\Delta\Theta'}^1(\Delta\theta) = \bigcup_{n=1}^{N_1} \{&\mu_{D_n}(d_1') \wedge \mu_{D_n}(d_2') \wedge \mu_{D_n}(d_3')\\
&\wedge \mu_{D_n}(d_4') \wedge \mu_{D_n}(d_5') \wedge \mu_{\Delta\Theta_n}(\Delta\theta)\} \quad (9)
\end{aligned}$$

where $\wedge$ denotes the minimum operation. In the same manner, when the inputs, ψ' and z' are given, the membership values of V' and $\Delta\Theta'$ for the goal-seeking behavior are calculated by

$$\mu_{V'}^2(v) = \bigcup_{n=1}^{N_2} \{\mu_{\Psi_n}(\psi') \wedge \mu_{Z_n}(z') \wedge \mu_{V_n}(v)\} \qquad (10)$$

$$\mu_{\Delta\Theta'}^2(\Delta\theta) = \bigcup_{n=1}^{N_2} \{\mu_{\Psi_n}(\psi') \wedge \mu_{Z_n}(z') \wedge \mu_{\Delta\Theta_n}(\Delta\theta)\}. \qquad (11)$$

D. Defuzzification of Output Variables

In order to determine the crisp output actions, $\overline{v}^k$ and $\Delta\overline{\theta}^k$, for the kth behavior from the fuzzy control actions, V' and $\Delta\Theta'$, a defuzzification process is required. When the method of the center of gravity is used as a defuzzification method, crisp control actions, in the case of discrete universe, are expressed by

$$\overline{v}^k = \left\{\sum_{n=1}^{p} v_n \mu_{V'}^k(v_n) \middle/ \sum_{n=1}^{p} \mu_{V'}^k(v_n)\right\} \qquad (12)$$

$$\Delta\overline{\theta}^k = \left\{\sum_{n=1}^{q} \Delta\theta_n \mu_{\Delta\Theta'}^k(\Delta\theta_n) \middle/ \sum_{n=1}^{q} \mu_{\Delta\Theta'}^k(\Delta\theta_n)\right\} \qquad (13)$$

where p and q denote the number of quantization levels of the fuzzy output actions, V' and $\Delta\Theta'$, in each behavior, respectively.

E. Behavior Rule Learning

In this section, an approach for constructing and tuning the fuzzy rules is described. When the control action signals are applied to mobile robot at time step, t the mobile robot moves

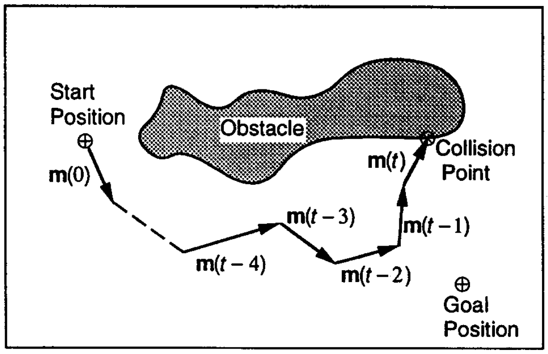


Fig. 7. The movement vectors associated with colliding with obstacle or moving away from goal position.

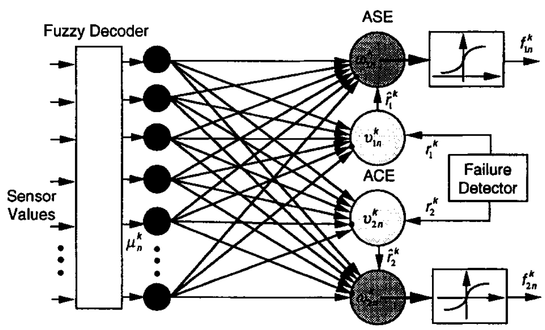


Fig. 8. The structure of neural network for learning the kth behavior.

in uncertain environment by the size of the movement vector, $\boldsymbol{m}(t)$ shown in Fig. 7. As the result of a series of movement vectors $\{\boldsymbol{m}(0), \cdots, \boldsymbol{m}(t-3), \boldsymbol{m}(t-2), \boldsymbol{m}(t-1), \boldsymbol{m}(t)\}$, the mobile robot may collide with obstacle or move away from goal position. To avoid such failures in case when the mobile robot navigates again through the environment similar to the previously experienced one, the rules contributed to generate the control action must be corrected. The rules influencing the mobile robot face with these failures, in turn, will be the rules used at previous time steps $t, t-1, t-2, \cdots$. Thus, such rules must be changed into the correct rules. This task is accomplished by two adaptive neuronlike elements [13] consisting of associative search element (ASE) and associative critic element(ACE) shown in Fig. 8. In order to accomplish the above process, Sutton and Barto's model [13] is used. Their model differs from Hebbian learning rule in that their model uses the associativity rather than the correlation of current inputs and outputs.

In our navigation system, in order to give associativity in learning the rules, the trace, $\overline{\mu}_n^k(t)$ of the fired nth rule is used. The trace of the rule at time step, $t+1$ is expressed by a weighted average of the membership value, $\mu_n^k(t)$ and the trace of the rule at time step, t. Therefore, the traces of all the rules can be calculated. The traces are prolonged for some period of time after t. The trace of the nth rule is then given by

$$\overline{\mu}_n^k(t+1) = \lambda\overline{\mu}_n^k(t) + (1-\lambda)\mu_n^k(t) \tag{14}$$

where superscript k denotes the kth behavior and $\lambda, (0 \leq \lambda \leq 1)$ is a trace decay rate. Here, $\mu_n^k(t)$ is calculated by

$$\mu_n^1(t) = \mu_{D_n}(d_1') \wedge \mu_{D_n}(d_2') \wedge \mu_{D_n}(d_3') \wedge \mu_{D_n}(d_4') \wedge \mu_{D_n}(d_5') \tag{15}$$

$$\mu_n^2(t) = \mu_{\Psi_n}(\psi') \wedge \mu_{Z_n}(z'). \tag{16}$$

As shown in Fig. 8, the ACE's receive the reinforcement signals, $r_m^k(t), (m = 1, 2)$ externally fed back from failure detector, respectively, and generate the internal reinforcement signals, $\hat{r}_m^k, (m = 1, 2)$, respectively. In case of avoidance behavior, the external reinforcement signals $r_m^k(t), (k = 1, m = 1, 2)$ are designed as follows:

$$\begin{cases} r_m^k = -1, 0 & \text{if } \min\{d_i | i = 1, 2, \cdots, 5)\} < R(1.0 + \epsilon) \\ r_m^k = 0.0 & \text{otherwise} \end{cases} \tag{17}$$

where ϵ denotes a safety factor set to 0.1 in this study. In case of the goal-seeking behavior, the external reinforcement signals $r_m^k(t), (k = 2, m = 1, 2)$ are designed as follows:

$$\begin{cases} r_1^k = +1.0 & \text{if } \psi(t-1) - \psi(t) > 0.0 \\ r_2^k = +1.0 & \text{if } \psi(t) < \pi/8 \\ r_m^k, (m = 1, 2) = 0.0 & \text{otherwise} \end{cases} \tag{18}$$

where $\psi, (-\pi < \psi \leq \pi)$ is the angle between heading direction of mobile robot and the direction of goal position with respect to the current position of mobile robot. In order to determine the internal reinforcement signal, $\hat{r}_m^k$ which plays a role of adjusting the weights of the ASE, predicting the external reinforcement is required. A prediction value, $p_m^k(t)$ of external reinforcement signal is calculated by using the temporal difference theory as follows:

$$p_m^k(t) = E\left\{ \sum_{t' \geq t}^{t'} \gamma^{(t'-t)} r_m^k(t'+t) \right\} \tag{19}$$

where $\gamma, (0 < \gamma < 1)$ is a positive constant determining the eventual extinction of prediction in absence of external reinforcement signal and $E\{\cdot\}$ denotes the expectation. In the case of correct learning, $p_m^k(t)$ from (19) is derived and expressed by

$$p_m^k(t-1) = r_m^k(t) + \gamma p_m^k(t). \tag{20}$$

However, in case of incorrect learning, (20) generates the error $\hat{r}_m^k(t)$, so called an internal reinforcement signal, defined as

$$\hat{r}_m^k(t) = r_m^k(t) + \gamma p_m^k(t) - p_m^k(t-1). \tag{21}$$

A prediction value, $p_m^k(t)$ is implemented as follows:

$$p_m^k(t) = G\left(\sum_{n=1}^{N_k} v_{mn}^k(t)\mu_n^k(t) \right) \tag{22}$$

where $G(\cdot)$ denotes a logistic function such as identity or sigmoid function. The logistic function used herein is a bipolar activation function expressed by $G(x) = 2/(1 + e^{-\xi x}) - 1$. In order for prediction value to be correct one, the weights of the ACE's must be updated. The weights of the ACE, $v_{mn}^k, (k = 1, 2, m = 1, 2, n = 1, 2, \cdots, N_k)$ are learned through the trace of the fired rule, $\overline{\mu}_n^k(t)$ and its output, $\hat{r}_m^k$. The subscripts m and n in v_{mn}^k denote the mth control action type and the nth weight strength of the ACE, respectively, and the superscript k denotes the kth behavior. The nth weight strength of the ACE associated with kth behavior and mth control action type is updated by

$$v_{mn}^k(t+1) = v_{mn}^k(t) + \beta \hat{r}_m^k(t)\overline{\mu}_n^k(t) \tag{23}$$

where β is a positive constant determining the rate of change of v_{mn}^k. In the same way as the weights of the ACE are updated, those of ASE are updated by

$$\omega_{mn}^k(t+1) = \omega_{mn}^k(t) + \alpha \hat{r}_m^k(t) e_{mn}^k(t) \tag{24}$$

where $\alpha, (0 < \alpha \leq 1)$ denotes a learning rate. Also, an eligibility trace of the nth rule for realizing the kth behavior is updated by

$$e_{mn}^k(t+1) = \rho e_{mn}^k(t) + (1-\rho) u_m^k(t) \mu_n^k(t) \tag{25}$$

where $\rho, (0 \leq \rho < 1)$ is a rate of decay. The eligibility trace indicates that certain rules have been used and what control actions are applied to the mobile robot at that time. Here, $u_m^k(t), (m = 1, 2)$ are the control actions for kth behavior and are defined as $\{u_1^k(t), u_2^k(t)\}^T = (v, \Delta\theta)^T$ where T denotes the transpose. Eventually, if the rules for each behavior are sufficiently learned in the specific uncertain environments, the weights of ASE will converge to the fixed values.

The tuning of the rules associated with each behavior is accomplished by the weights of the ASE. As shown in Figs. 6(a), (b) and (c), the input linguistic values are completely known in the universe of discourse before learning. On the other hand, the membership functions of the output linguistic values shown in Figs. 6(d) and (e) are partially known only in their shape, but their center values are yet unknown in the universe of discourse. Thus, the center value of the membership function of the output linguistic value used in nth rule is determined at each time step by the nth weight of the ASE associated with mth control action type. As can be seen from (24) and (25), when all the control actions are zero, the weights of the ASE can not be updated. Therefore, in this study, the center values of the output fuzzy set representing the linear velocity are initially determined in such a way that they have always nonzero values. The center values, $f_{mn}^k(\omega_{mn}^k(t), t), (m = 1, 2, k = 1, 2, n = 1, 2, \cdots, N_k)$ at time step t are calculated by

$$f_{mn}^k(\omega_{mn}^k(t), t) = \begin{cases} \dfrac{\omega_{mn}^k(t) F_{\max}^k}{K_m \max_l(|\omega_{ml}^k(t)|) + \omega_{mn}^k(t)} + g_m \\ \quad \omega_{mn}^k(t) \geq 0 \\ \dfrac{\omega_{mn}^k(t) F_{\max}^k}{K_m \max_l(|\omega_{ml}^k(t)|) - \omega_{mn}^k(t)} + g_m \\ \quad \omega_{mn}^k(t) < 0 \end{cases} \tag{26}$$

where $F_{\max}^k$ is a positive constant determining the range of the center value associated with kth behavior. The k_m is the slope of the membership function of the mth control action. Here, the offset values, g_1 and g_2 are set to a positive constant and zero, respectively. In the above equation, in order for the output fuzzy sets to be within the universe of discourse, the largest absolute value among the weights of the ASE is used to determine the center values. Once all the membership functions of the output fuzzy sets are determined through reinforcement learning, the rule base of each behavior is completely tuned.

TABLE I
THE PARAMETER VALUES USED FOR SIMULATION STUDY

$\alpha = 0.8$	$F_{\max}^1 = 15.0$cm / sec
$\beta = 0.8$	$F_{\max}^2 = \pi / 2$ rad
$\gamma = 0.95$	$\chi = 0.0003$
$\rho = 0.85$	$K_r = 0.0003$
$\lambda = 0.5$	$a_{\max} = 0.01$ m^2 / sec
$\xi = 1.5$	$g_1 = 20.0$ cm / sec
$K_m = 0.1$	$g_2 = 0$

F. Behavior Selector

In order for mobile robot to arrive at the goal position without colliding with obstacles, appropriate behavior must be selected according to situation around the mobile robot. When the obstacles are located around the mobile robot as shown in Fig. 2, the mobile robot must reduce its speed in order not to collide with them. If the maximum acceleration, $a_{\max}$ is applied to the mobile robot with an arbitrary initial velocity, the minimum time required to avoid the obstacle is calculated by $v \cos \varphi_j / a_{\max}$. But, if a constant acceleration less than the maximum value is applied to the mobile robot, the time required to avoid the obstacle is calculated by $2(e_j - R)/v \cos \varphi_j$. Therefore, when the distance from the center of mobile robot to the obstacle and the velocity component towards the obstacle are given, the marginal avoidance time can be defined as the difference between the above two required times. Since the marginal avoidance time denotes the degree of collision, in this study it is used to represent the repulsive potential. Repulsive potential, E_{rep} is produced by the inversion of the marginal avoidance time being the function of mobile robot velocity and the distances to obstacles measured by ultrasonic sensor array. Thus, a repulsive potential is defined as

$$E_{\text{rep}} = \sum_j^{S_N} \frac{K_r a_{\max} v \cos \varphi_j}{2a_{\max}(e_j - R) - v^2 \cos^2 \varphi_j} \tag{27}$$

where K_r is a scale factor and S_N is the number of ultrasonic sensors. Repulsive potential increases as the mobile robot gets closer to the obstacles and approaches zero as its velocity in the direction of obstacle approaches zero.

When the mobile robot configuration is given in Fig. 2, the attractive potential is calculated by the minimum time to

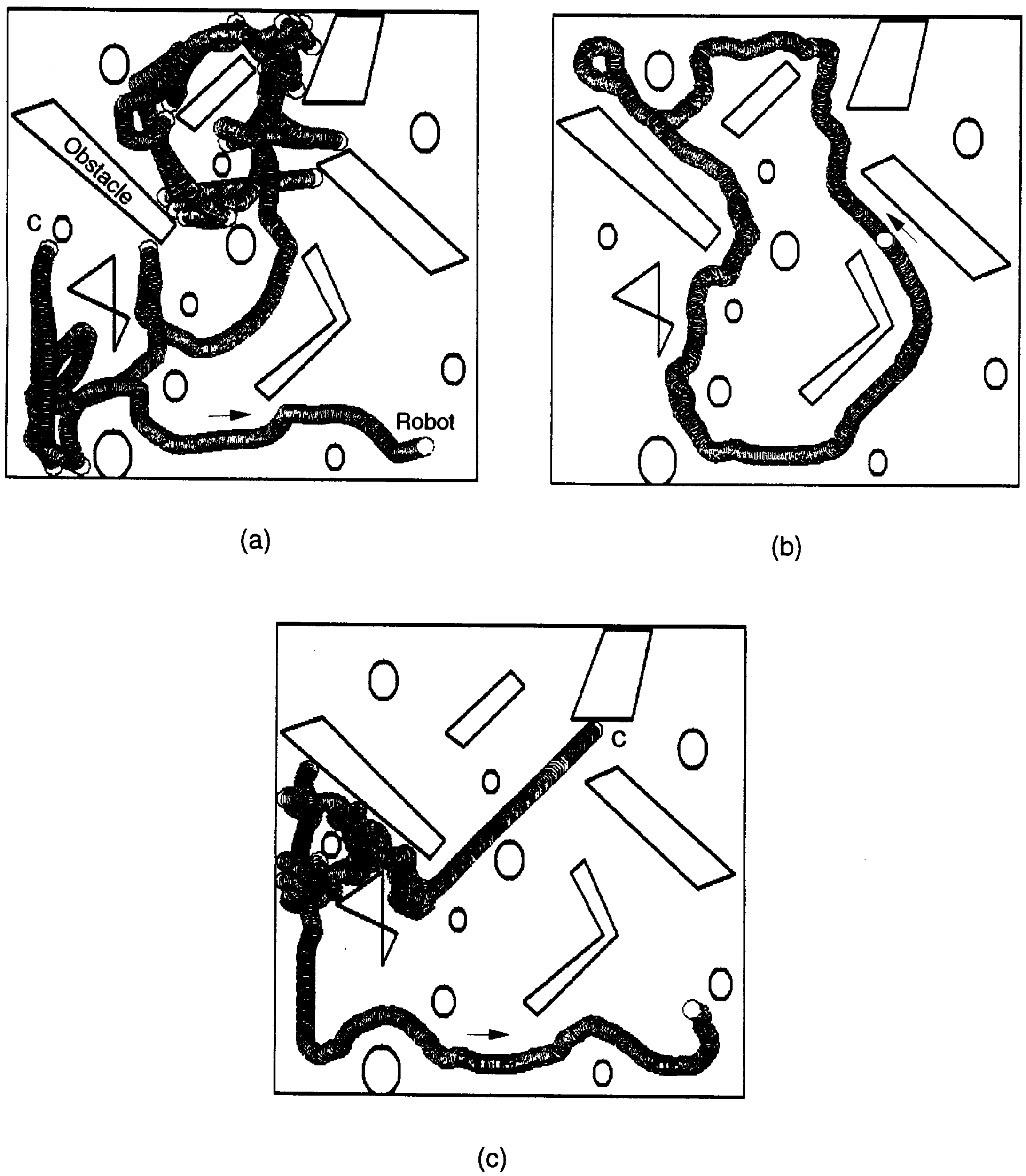


Fig. 9. The trajectory of the mobile robot in learning process: avoidance behavior.

transfer the mobile robot with initial velocity, v to the goal position. This potential is calculated using a single switching time as follows:

$$E_{\text{att}} = \frac{\sqrt{2v^2 \cos^2 \psi + 4a_{\max} \parallel \boldsymbol{X} - \boldsymbol{X}_g \parallel} - v \cos \psi}{a_{\max}}. \tag{28}$$

Attractive potential depends on the distance to goal position, heading angle difference and initial velocity. As can be seen from (28), attractive potential increases as the mobile robot moves away from goal position or as its heading angle difference gets larger.

Situation around the mobile robot can be represented by use of the sum of two potentials, $E = E_{\text{rep}} + E_{\text{att}}$. In order that the mobile robot can go through environment to the goal position without colliding with obstacles, one of two behaviors must be selected. When two behaviors are assumed to be used as behavior at next step, the control actions derived from each behavior are calculated. Then, the changes of the potential, $c_a = (\Delta E)_{\text{avoidance}}$ and $c_g = (\Delta E)_{\text{goal}}$, resulted from the

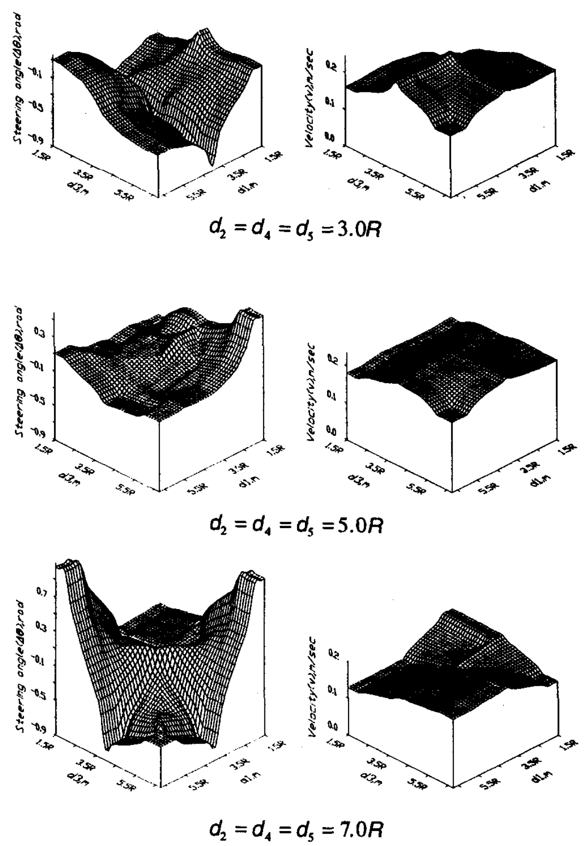


Fig. 10. The action surfaces of the avoidance behavior with respect to the variation of d_1 and d_3.

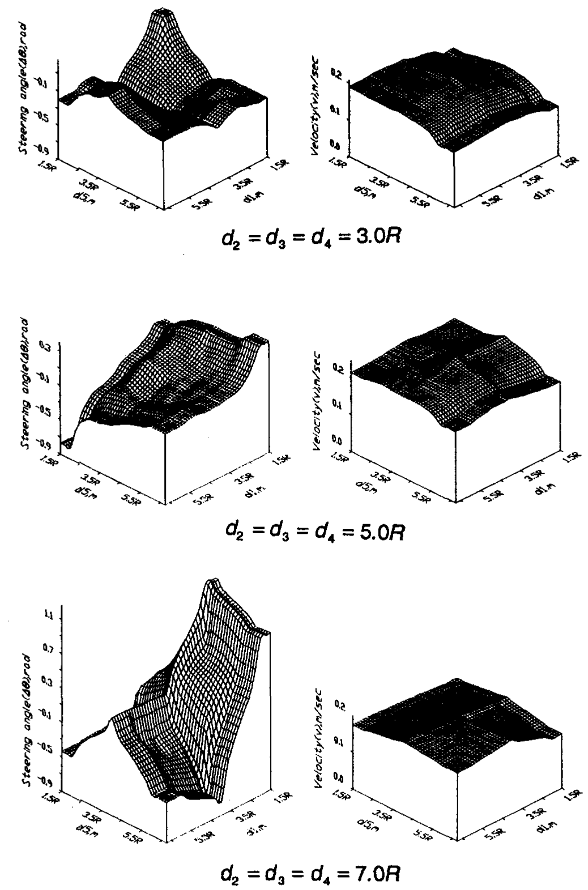


Fig. 11. The action surfaces of the avoidance behavior with respect to the variations of d_1 and d_5.

actions of two behaviors are calculated, respectively. Based on the changes of the potential, one behavior to be used at next action step is chosen by the behavior selector. To determine this, behavior selector uses the switching function which is a bistable recursive one for smooth switching between two behaviors at each action step. A decision or switching, Π is defined by

$$\Pi(t) = \Gamma(t)f(c_a) - (1 - \Gamma(t))f(c_g) \tag{29}$$

where $\Gamma(t), (0 < \Gamma(t) < 1)$ is the time-varying weighting factor used to avoid the case when the changes of potential are equal. In the above equation, $f(x)$ is a positive and monotonically increasing function defined as $e^{0.5x}$. The $\Gamma(t+1)$ at time step $t+1$ is calculated by using an unipolar logistic function, $(1+e^{-\chi\Pi(t)})^{-1}$ where χ is a positive constant determining the stability of switching. Thus, if $\Pi < 0$, the avoidance behavior is selected and if $\Pi > 0$, the goal seeking behavior is selected.

IV. Simulation and Results

A. Simulation Conditions

As an illustration of the foregoing process, a series of simulations were made using an arbitrary constructed environment composed of sixteen obstacles. It was assumed that the radius of the mobile robot is 0.20 m and the size of environment is 10 m $\times$ 10 m. The maximum speed of the mobile robot was assumed to be 35 cm/sec. The measurement range of the sensor was assumed to be eight times the radius of the mobile robot. In order to acquire information about environment, a sensor simulator which outputs the range data of ultrasonic sensors was used. For the sensor simulator, it was assumed that the ultrasonic sensors can ideally detect the nearest obstacles within their beams. The rule base is constructed for each behavior to infer two actions such as incremental steering angle and linear velocity. In the simulation, the rule bases for avoidance and goal-seeking behaviors consist of 2×35 and $2 \times 4 \times 7$ rules, respectively. In the beginning, the center values of the membership functions of the output fuzzy sets are fixed to constant values, g_1 and g_2 listed in Table I, respectively. Their center values approach correct values as the learning step increases. All the parameters used in the simulation for reinforcement learning are given in Table I.

B. Simulation Procedure for Determination of Control Actions

By use of (2) and (3), the eighteen distance values measured by sensors are grouped into the five distance values belonging

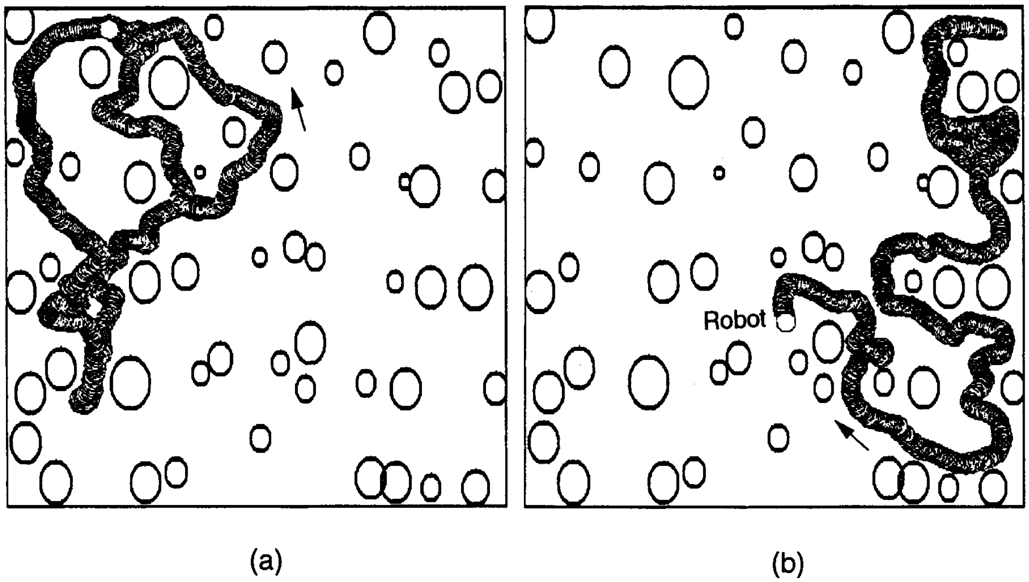


Fig. 12. The example of the avoidance behavior: different environments.

to each sensor suit. These five distance values are fed into a fuzzy decoder which partitions the mobile robot's input space into some subspaces. The number of the partitioned subspaces becomes the number of the rules for realizing each behavior and the subspace represents a situation around the mobile robot. Thus, when the distance values are given, the degree to which the mobile robot belongs to the situation is represented by membership values expressed by (15) and (16) which correspond to the premise of the rule. If the center values of the membership functions of output fuzzy sets are given by the rule learning process, fuzzy actions to be applied to the mobile robot are determined by (6) through (11) and crisp actions are determined by (12) and (13).

C. Simulation Procedure for Rule Learning

The rule learning process begins from initialization of the variables associated with this process. In the beginning, the initial weights of the ACE are set to small nonzero values, while the initial values of other variables are set to zero. The situation where the mobile robot is at present instant can be determined by (15) and (16). In this case, some rules with nonzero value of membership function exist. Using these membership values, the traces of the rules are calculated. The traces of such rules expressed by (14) increase while the traces of rules with zero value of membership function decrease. Thus, the rules with nonzero trace value cause the mobile robot to be faced with present situation. Using these traces, the center values of the membership functions of output fuzzy sets are adjusted so that the failures do not occur. In the simulation, failures are assumed to occur when mobile robot collides with obstacles or moves away from the goal position. The failure detector signals expressed by (17) and (18) are determined by designer and the characteristics of each behavior depends on the type of failure detector. In order to adjust center values so that the failures do not occur, prediction of failures using (19) through (22) is performed in ACE. Then, the ACE generates the internal reinforcement signal determined by (21) and their weights are updated by (23). Once the internal reinforcement signal is determined, the weights of ASE directly associated with the center values of the membership functions of output fuzzy sets are updated by (24) and (25). Using these weights, the center values are determined by (26).

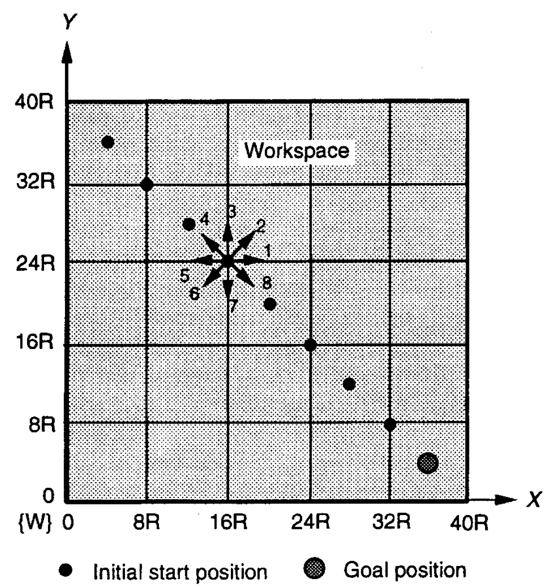
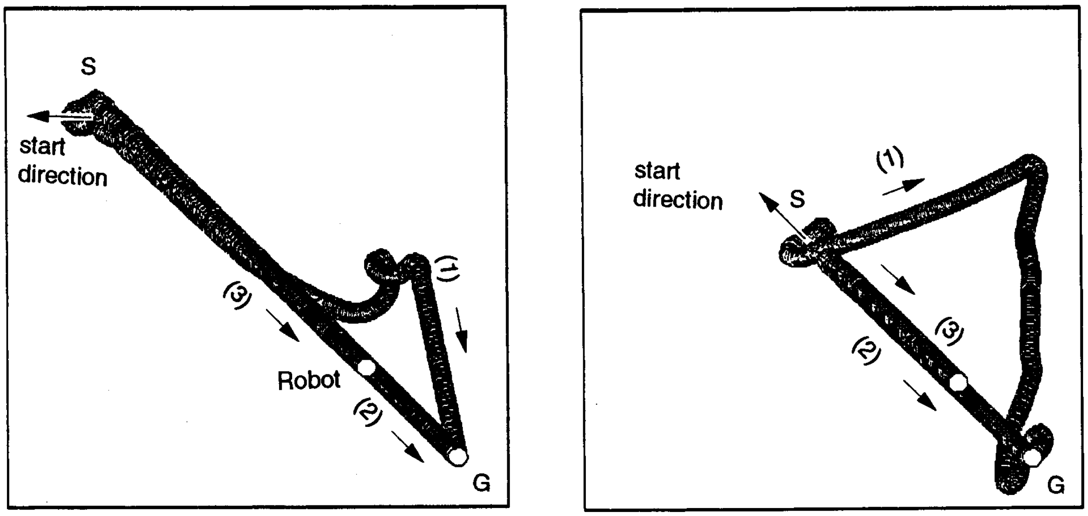


Fig. 13. The eight start positions with eight heading angles and a goal position for training the goal-seeking behavior.



(a) (b)

Fig. 14. The trajectory of the mobile robot in learning process: goal-seeking behavior.

D. Simulation for Learning the Avoidance Behavior

Let us consider the case of learning the avoidance behavior using the above procedure. Since the center values of the output fuzzy sets representing the linear velocity are initially set to nonzero value, the mobile robot begins to move in the direction of initial heading angle and the rule learning is gradually achieved. The first trial for learning the rules continues until the mobile robot collides with obstacles: A trial consists of a series of learning steps until a collision occurs. Thus, if the collision occurs, the first trial stops at that time. To get away from the collision region, the mobile robot may choose many different paths. But, the most efficient way of selecting the next initial position and heading angle is to utilize the information on the positions and heading angles at the previous time steps. Here, for safety reasons, the mobile robot is assumed to choose the information used at time $t-2$ which indicates two time steps behind the current time t. By use of the information, the second trial begins, that is, its start position and heading angle are set to the position and direction opposite to the heading angle at time step $t-2$. Simultaneously, the weights of ASE learned just before the failure are used for the second trial and the values of other variables are set to zero. As shown in the upper part of Fig. 9(a), the mobile robot frequently collides with obstacles at the position marked as C. The number of learning steps is relatively small until a collision occurs. But, the number of collision begins to decrease as the number of learning steps increases as shown in the lower part of the figure. The above learning process continues until the mobile robot visits, as possible, all the regions of the specified environment.

As the learning proceeds, in some cases the mobile robot may be trapped. In such cases, a new start position and a heading angle must be given in order to learn other situations. The trajectory of the mobile robot after the number of total learning steps gets larger than 35,000 is shown in Fig. 9(b).

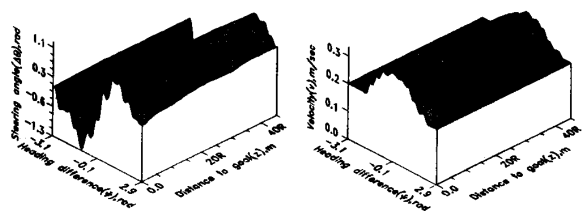


Fig. 15. The action surfaces of the goal-seeking behavior.

The mobile robot successfully navigates, and maintains a safe distance from obstacles. This means that the mapping between input state space and output action space is correctly established.

In the above simulation, we assumed that all the rules associated with the avoidance behavior are completely unknown. This assumption was made in order to show the powerfulness of learning capabilities of the reinforcement learning method. However, some rules may be roughly determined by using expert's knowledge and then they can be tuned by reinforcement learning method. In this case, we can improve the performance of the learning system. Fig. 9(c) shows the trajectory of the mobile robot in the case when 27 rules among 243 rules consisting of rule base of avoidance behavior are initially given based on expert's knowledge. The 27 rules are associated with the cases when IF-parts of the rules, that is, "VN, NR, FR, NR, VN" are symmetric with respect to the center linguistic value (FR). As can be seen from Fig. 9(c), the number of collision is smaller than that of collision in Fig. 9(a). If a priori knowledge is used in initial learning stage, the performance of the learning system can be greatly improved.

The action surfaces representing the mapping relation between the two spaces are shown in Figs. 10 and 11. The action surfaces are obtained assuming that all the rules are completely unknown. Fig. 10 shows the action surfaces with

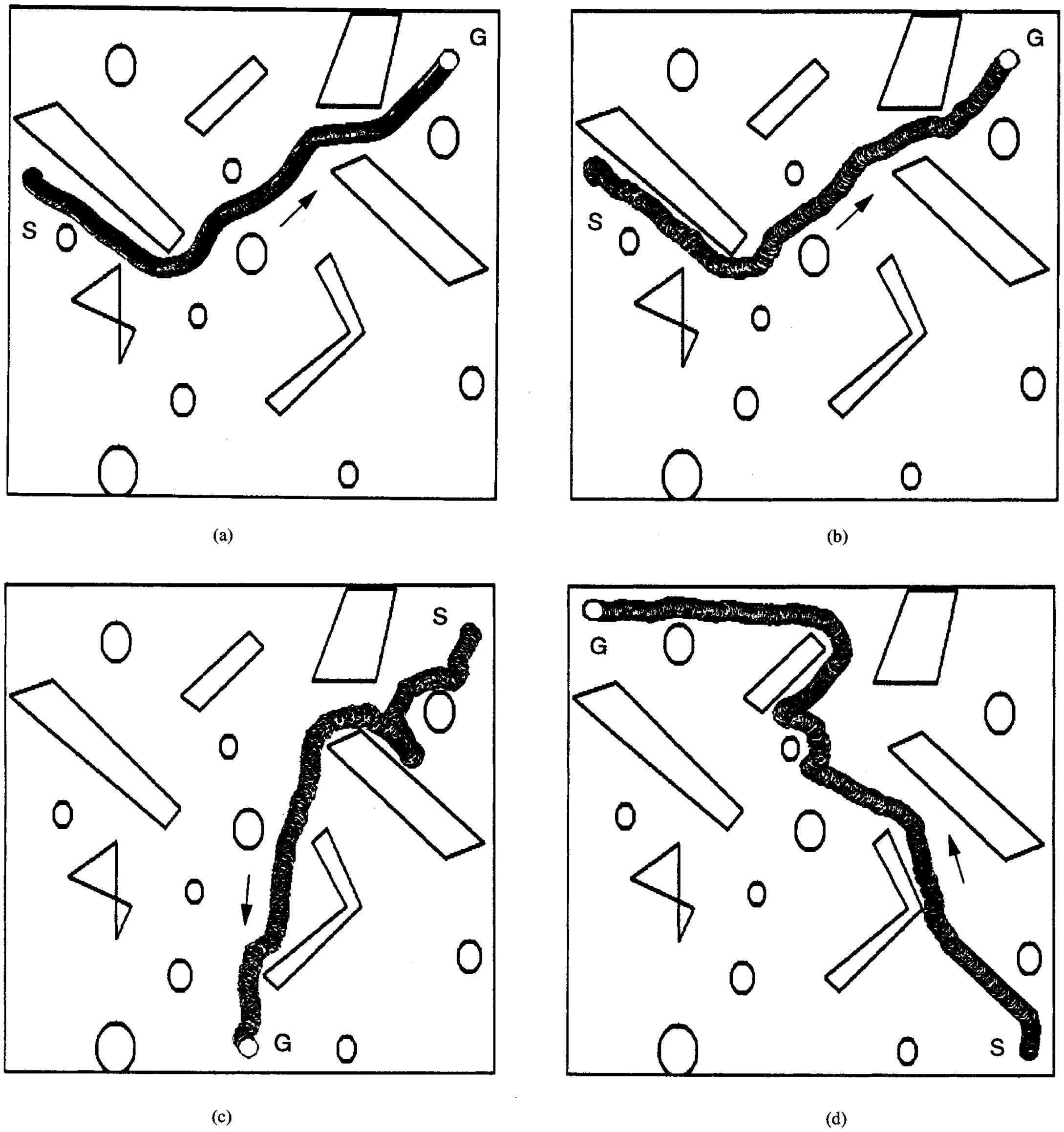


Fig. 16. The illustrations of the proposed navigation.

respect to the variations of the positions of the obstacles located in front of the first and third sensor suits, ss_1 and ss_3. In this case, the obstacles in front of other sensor suits ss_2, ss_4 and ss_5 are assumed to be stationary and located at $3.0R, 5.0R$ and $7.0R$, respectively. Fig. 11 shows the action surfaces with respect to the variations of the positions of the obstacles located in front of the first and fifth sensor suits, ss_1 and ss_5. These actions make the mobile robot move into a collision-free region. From these figures, we can decide if sufficient learning is accomplished or not. If sufficient learning is done, the steering angle and linear velocity of mobile robot in action surfaces should not be simultaneously equal to their respective initial values. In order to test the adaptability of learning system, a simulation was performed. As can be seen from Fig. 12, although the rule bases learned in environment shown in Fig. 9 are used, the mobile robot successfully avoids the obstacles in a new environment.

E. Simulation for Learning the Goal-Seeking Behavior

In order to learn the rule base for goal seeking behavior, 8 initial positions and 8 initial heading angles of mobile robot shown in Fig. 13 are used. Figs. 14(a) and (b) show the learning processes when the initial posture of mobile robot, that is, its start position and heading angle with respect to the frame $\{W\}$, are given as $(4R, 36R, 180°)$ and $(20R, 20R, 135°)$, respectively. The goal position, G, is fixed

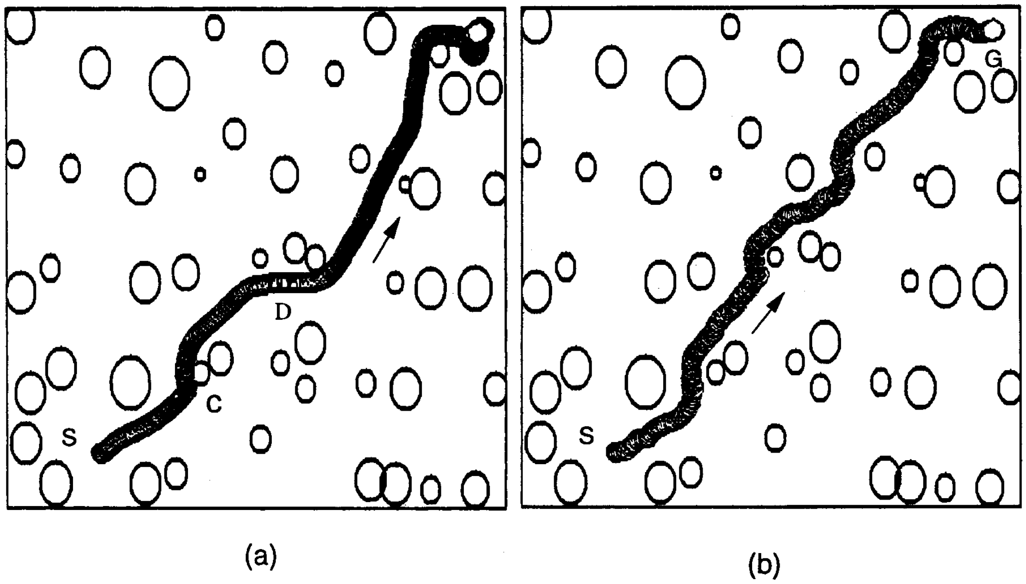


Fig. 17. The illustrations of the navigation: different environments.

at position, $(36R, 4R)$. When the mobile robot arrives at the goal position, the first trial(1) for learning the rules is completed. Then, the second(2) and third trials(3), in turn, begin again from a priori given initial position and heading angle with the weights of ASE learned in the previous one. These processes repeat until the tenth trial is completed. In these learning processes, the initial values of other variables were set to zero as in the case of the avoidance behavior learning. As shown in the figures, the number of the action steps from start position to the goal position decreases as the number of trial increases. From Fig. 14, the rules for goal-seeking behavior are almost learned by the first trial. After the learning processes with regard to all the combinations of the start and heading angle as shown in Fig. 13 are completed, the action surfaces which represent the mapping relation between the two spaces can be drawn. Fig. 15 shows the action surfaces obtained for the goal-seeking behavior. The linear velocity of mobile robot decreases irrespective of the distance to goal position as the heading angle difference increases. When $\psi = 0$, the mobile robot moves toward the goal position with a maximum velocity. It is also noted that the absolute values of the incremental steering angle when $\psi > 0$ are similar to the case when $\psi < 0$, but their signs are opposite to each other.

F. Simulation for Combining Two Behaviors

In this section, the proposed method is compared with a force field method [6], [7]. In the force field method, obstacles exert repulsive forces onto the mobile robot, while the goal point gives an attractive force to the mobile robot. The sum of all forces, that is, resultant force, determines the subsequent steering angle and linear velocity of the mobile robot. Fig. 16(a) shows the navigation result when the force field method is applied to the mobile robot. As can be seen from this figure, the mobile robot can smoothly navigate through environment. In this simulation, two force constants were well tuned so that the mobile robot may navigate through environment without colliding with obstacles. These force constants will be used in the navigation in a new environment and, at that time, the performance of the force field method with these constants will be compared with that of the proposed method.

In the proposed method, once the rule bases for the two behaviors are completely built through reinforcement learning, the two behaviors will be combined so that the mobile robot arrives at the given goal position without colliding with obstacles. When the mobile robot navigates in a certain environment, one of these behaviors must be selected at each action step in order to accomplish its goal. In this study, the repulsive and attractive potentials are used to select an appropriate behavior. This is performed by the action of the bistable recursive switching function expressed by (29). To test the effectiveness of the proposed method using the behavior selector, different start and goal positions and its initial heading angles are used. Fig. 16(b) shows a case when a local minimum does not occur. In this case, the mobile robot located at start position, S, arrives at the goal position, G, without colliding with obstacles. Figs. 16(c) and (d) show a case when a local minimum occurs. From the figures, even if the mobile robot is faced with the situation where a local minimum occurs, the proposed navigator designed by using the behavior selector shows an ability to escape from a local minimum.

To see a capability adapting to a new environment, a series of simulations using the two methods was performed under the same obstacle configuration, start and goal positions. In this simulation, two force constants and rule bases tuned for the previous simulation were used. Fig. 17(a) shows the navigation results when the force field method was applied to the mobile robot. As shown in the figure, the mobile robot collides with obstacle at the position marked as C. Also, at the position marked as D, the mobile robot stops because the sum of the two forces becomes zero. In this case, we applied an additional small force to the mobile robot such that the mobile robot can escape from a local minimum. Furthermore, when the goal position is located near the obstacles, the mobile robot arrives at the goal position after some wandering. Fig. 17(b) shows the navigation results of the proposed method. On the contrary to the case of the previous method, the mobile robot navigates without colliding with obstacles. These results indicate that the proposed method has better ability to adapt to a new environment than the force field method does.

V. Concluding Remarks

A sensor-based navigation method utilizing the fuzzy logic and reinforcement learning has been presented for complex environments. The proposed navigator consists of avoidance and goal-seeking behaviors. These are independently designed to accommodate complex environments and combined by the behavior selector which uses a bistable switching function. Reinforcement learning is used to construct the rule bases for two behaviors in such a way that the correct actions can be acquired in unknown environmental situations with which the mobile robot is faced at any instant. In order to show the effectiveness of the proposed method, a comparison with a force field method has been performed under the same obstacle configuration. The proposed method enables the mobile robot to navigate through complex environment where a local minimum occurs and to adapt to new environments. Since the rule bases are constructed in learning-by-observation manner, the two behaviors can be autonomously realized. In this study, it was assumed that the mobile robot's input space for avoidance behavior has the five dimensions by introducing the sensor suits. In general, the higher the input dimension is the heavier the computational burden will result in.

References

[1] T. Lozano-pérez and M. A. Wesley, "An algorithm for planning collision-free paths among the polyhedral obstacles," *Communications of the ACM* , vol. 22, no. 10, pp. 560–570, Oct. 1979.

[2] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," *Int. J. of Robotics Research*, vol. 5, no. 1, pp. 90–98, Spring 1986.

[3] O. Takahashi and R. J. Schilling, "Motion planning in a plane using generalized Voronoi diagrams," *IEEE Trans. Robotics Automat.*, vol. 5, no. 2, pp. 143–150, 1989.

[4] H. R. Beom and H. S. Cho, "Path planning for mobile robot using skeleton of free space," in *Proc. IFAC Symp. Information Control Problems for Manufacturing* (Toronto, Canada, May 25–28 1992), pp. 509–513.

[5] R. A. Brooks, "A robust layered control system for a mobile robot," *IEEE Trans. Robotics Automat.*, vol. RA-2, no. 1, pp. 14–23, Mar. 1986.

[6] J. Borenstein and Y. Koren, "Real-time obstacle avoidance for fast mobile robot," *IEEE Trans. Syst. Man Cyber.*, vol. 19, no. 5, pp. 1179–1187, Sept./Oct. 1989.

[7] J. Borenstein and Y. Koren, "Potential field methods and their inherent limitations for mobile robot navigation," in *Proc. IEEE Int. Conf. Robotics and Automation* (Sacramento, CA, Apr. 9–11, 1991), pp. 818–823.

[8] T. Takeuchi, "An autonomous fuzzy mobile robot," *J. Robotics Soc. Japan*, vol. 6, no. 6, pp. 549–556, Dec. 1988.

[9] S. Ishikawa, "A method of indoor mobile robot navigation by fuzzy control," in *Proc. Int. Conf. Intelligent Robots and Systems* (Osaka, Japan, Nov. 3–5, 1991), pp. 1013–1018.

[10] C. Kozakiewicz and M. Ejiri, "Neural network approach to path planning for two dimensional robot motion," in *Proc. Int. Conf. Intelligent Robots and Systems* (Osaka, Japan, Nov. 3–5, 1991), pp. 818–823.

[11] M. Sekiguchi, S. Nagata and K. Asakawa, "Mobile robot control by a structured hierachical neural network," *IEEE Control Systems Mag.*, vol. 10, no. 3, pp. 69–76, Apr. 1990.

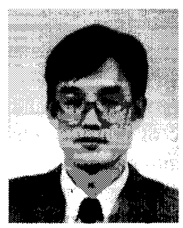
[12] H. R. Beom and H. S. Cho, "A sensor-based obstacle avoidance controller for a mobile robot using fuzzy logic and neural network," in *Proc. Int. Conf. Intelligent Robots and Systems* (Raleigh, NC, July 7–10, 1992), pp. 1470–1475.

[13] A. G. Barto, R. S. Sutton, and C.W. Anderson, "Neuronlike adaptive elements that can solve difficult learning control problems," I *IEEE Trans. Syst. Man Cyber..*, vol. SMC-13, no. 5, pp. 834–846, Sept./Oct. 1983.

[14] C. C. Lee, "Modeling the behavioral substrates of associate learning and memory: Adaptive neural models," *IEEE Trans. Syst. Man Cyber..*, vol. 21, no. 3, pp. 510–520, May/June 1991.

[15] Polaroid Corporation, *Ultrasonic ranging system* , 1988.

[16] C. C. Lee, "Fuzzy logic in control systems:fuzzy logic controller, part II," *IEEE Trans. Syst. Man Cyber..*, vol. 20, no. 2, pp. 419–435, Mar./Apr. 1990.



Hee Rak Beom was born on March 27, 1959, in Kwangju, Korea. He received B.S. degree in Mechanical Engineering from Korea University, Seoul, Korea in 1982 and M.S. degree in Production Engineering from Korea Advanced Institute of Science and Technology (KAIST), in 1985, where he is currently working toward Ph.D. degree in Department of Precision Engineering and Mechatronics.

He has joined in development of industrial robot control system at Gold Star Industrial Systems, Co., Ltd., since 1985. His research interests include mobile robot navigation, path planning, obstacle avoidance, intelligent control applications and sensors for robotic applications.

Hyung Suck Cho was born on October 8, 1944, in Seoul, Korea. He received B.S. degree in Industrial Education from Seoul National University, Seoul, Korea, M.S. degree in Mechanical Engineering from Northwestern University, Evanston, IL, and Ph.D. degree in Mechanical Engineering from the University of California at Berkeley, CA, in 1971, 1973, and 1977, respectively.

From 1977 to 1978 he was a postdoctoral fellow with the Department of Mechanical Engineering, University of California at Berkeley. Since 1978 he has been a Professor with the Department of Production Engineering, Korea Advanced Institute of Science and Technology, Taejon, Korea. From 1984 to 1985 he was a visiting scholar at the Institute für Productionstechnik und Automatisierung (IPA), Stuttgart, Germany, where his research topic was robot-based assembly. In 1987, 1992, he was a visiting Scholar at Ritsumeikan University, Japan and at the University of Paderborn, Germany. From 1990 to 1993, he served as a vice chairman of Manufacturing Committee, IFAC and now is one of the members of Technology Committee. Currently, he is serving as one of the editorial board members for international journals, *Robotica, IFAC Control Engineering Practice*, and *Advanced Robotics*. His research interests include robotics and automation, intelligent control applications, manufacturing process control, machine vision, and robotic assembly. In these research areas, he published over 140 papers in international journals and conferences and 40 papers in Korean journals.

Dr. Cho is a member of ASME, SME, KSPE, and KSME.