Paper

Navigation of Autonomous Mobile Robot under Decision-making Strategy tuned by Genetic Algorithm

Fei Wang* Student Member
Takuya Kamano* Member
Takashi Yasuno* Member
Takayuki Suzuki* Member
Hironobu Harada* Member

This paper describes a novel application of genetic algorithm for navigation of an autonomous mobile robot (AMR) under unknown environments. In the navigation system, the AMR is controlled by the decision-making block, which consists of neural network. To achieve both successful navigation to the goal and the suitable obstacle avoidance, the connection weights of the neural network and speed gains for predefined actions are encoded as genotypes and are tuned simultaneously by genetic algorithm so that the static and dynamic danger-degrees, the energy consumption and the distance and direction errors decrease during the navigation. Experimental results demonstrate the validity of the proposed navigation system.

Keywords: AMR, Navigation, Obstacle avoidance, Decision-making, Genetic algorithm, Danger-degree

1. Introduction

The acquisition of navigation competency is one of the key problems in the development of autonomous mobile robots (AMRs). To solve this task efficiently, several methods have been investigated. These methods are roughly classified into the behavior-based approach, the robot learning approach and the evolutionary approach.

In behavior-based systems, behavior is achieved through coordination of a set of purposive perception-action units, called behaviors. Based on carefully selected sensory information, each behavior produces commands to control the robot with respect to a well defined aspect of the overall task. Usually, these systems need a coordination mechanism for actions selection, which can be modeled by neural network approach (1), the subsumption architecture (2), a method based on Bayesian decision theory (3) and fuzzy rules (4), etc.

The robot learning is based on the idea that a control system (typically a neural network) can be trained using uncompleted data and then allowed to rely on it ability to generalize the acquired knowledge to novel conditions. The different learning algorithm has been used for various purpose: i.e. back-propagation learning (5); reinforcement learning (6); classifier system (7); Knhonen Self-Organized Maps (8), etc.

The evolutionary robotics approach which shares many common characteristics with methods outlined above is a way to develop robots and their sensor-motor control system through an automatic design process involving Artificial Evolution (9)~(11). The evolutionary approach is usually based on genetic algorithm. An initial population with the different "genotypes", which encodes the control system of an AMR, is created randomly. Each robot is evaluated by a score (fitness) that measures its ability to perform the desired task in the environment. Those individuals that have obtained higher fitness values are allowed to reproduce by generating copies of the genotypes with the addition of changes introduced by some genetic operators (e.g. mutation, crossover, etc.). By repeating this process for generations, optimal genotype individual can be obtained in the final.

In this paper, the evolutionary approach is introduced to obtain the suitable navigation for an AMR under unknown environments. In the navigation system, a decision-making block plays an important role to select the suitable action from the six fundamental actions during navigation. To realize reasonable decision-making of the AMR, the static and dynamic danger-degrees (12) for the obstacle are introduced. The danger-degrees represent the instinctive human knowledge for environment recognition and are calculated by using the fixed fuzzy rules.

The decision-making block consists of a neural network, of which connection weights are encoded as part of genotype and are tuned by genetic algorithm so that the danger-degrees for obstacle, the energy consumption and the distance and direction errors decrease during the navigation to the goal. After tuning process, the decision-making block can achieve the suitable avoidance action from the obstacle and the successful

* The University of Tokushima
Dept. of Electrical and Electronic Engineering, Faculty of Engineering
2-1, Minamijosanjima, Tokushima 770-8506

navigation to the goal.

The rest of this paper is organized as follows. Section 2 describes the configuration of the experimental system. Section 3 introduces the overall control system. The off-line tuning scheme is represented in section 4. The simulation and experimental results are demonstrated in section 5. Conclusions are addressed in the last section.

2. Experimental Setup

2.1 Tested AMR Fig.1 shows the tested AMR on which two driving wheels driven by dc servo motors and eight ultrasonic sensors are mounted. The speed and travel direction are controlled by the dc servo motors of two drive wheels. The eight ultrasonic sensors can detect the surrounding obstacles. Two white bowls on the top of the AMR are the marks to detect the current position and direction of the AMR by a CCD camera.

2.2 Experimental Configuration The hardware configuration of experimental setup is shown in Fig.2. It is assumed that a rectangular arena of 1.8 × 1.4 m corresponds to the experimental field in which three dynamic obstacles are included. Each obstacle is a cylindrical form with a diameter of 0.08 m and a height of 0.20 m. The obstacle can move straightly with the constant speed of 0.15 m/s.

The control signals to the right and left wheels are calculated by means of computer software written in Visual C++ language, and delivered to motor drivers through a 12-bit digital-to-analog (D/A) converter. The positions of obstacles on the robot-centered coordinate are detected by ultrasonic sensors and those of the AMR on the reference coordinate system are measured by CCD camera with image processor. The measured positions are fed to host computer by a 12-bit analog-to-digital (A/D) converter as the environment recognition information. The sampling interval is set to 200 ms.

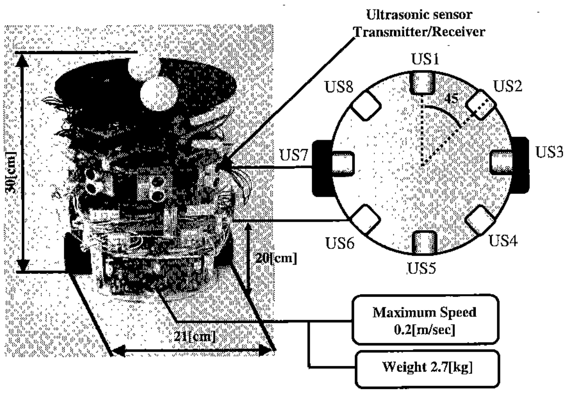


Fig. 1. Tested autonomous mobile robot.

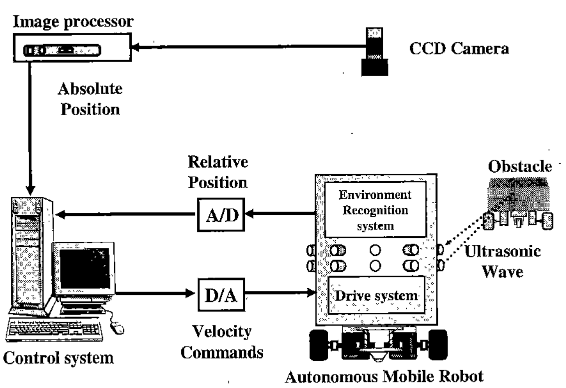


Fig. 2. Experimental configuration.

3. Actual Control System of AMR

Fig.3 shows the proposed overall control system. The system acts as an obstacle avoidance motion controller adding to a feedback controller to the goal.

3.1 Distance and Direction Errors Estimator In this block, the distance and the direction errors between goal and current position of the AMR are estimated and are fed to a decision-making block. The distance and the direction errors are given by

$$d_{error} = S\left(\sqrt{(x_g - x)^2 + (y_g - y)^2}\right) \cdots\cdots\cdots (1)$$

$$\begin{cases} \Psi_{error} = S(\Phi - \theta) \\ \theta = \arctan \dfrac{y_g - y}{x_g - x} \end{cases} \cdots\cdots\cdots\cdots\cdots\cdots (2)$$

$$S(x) = \left| \frac{2}{1 + \exp(-x)} - 1 \right| \cdots\cdots\cdots\cdots\cdots (3)$$

where (x_g, y_g) is the goal position, (x, y) is the current position of AMR measured by the CCD camera, Φ is the travel direction of the AMR, θ is the goal direction. S is the normalization function which maps the real values into the interval [0,1]. Notice that the distance and the direction errors are used not only as the feedback signals but as the estimation signal in the tuning block.

3.2 Decision-making Block In the proposed control scheme, the decision-making block is the most important part, and decides the action of the AMR. The decision-making block consists of a neural network shown in Fig.4. The input signals are the distance and direction errors to the goal, the current and one sampling delayed outputs of the ultrasonic sensors. The outputs of the block corresponds to six fundamental actions for the AMR, that is **G**o **S**traight, **R**ight **T**urn action, **L**eft **T**urn action, **R**ight **Ro**tation action, **L**eft **Ro**tation action and **ST**op action. Only one action that has the highest output level in the output layer is selected as the suitable action. In addition to the input and output

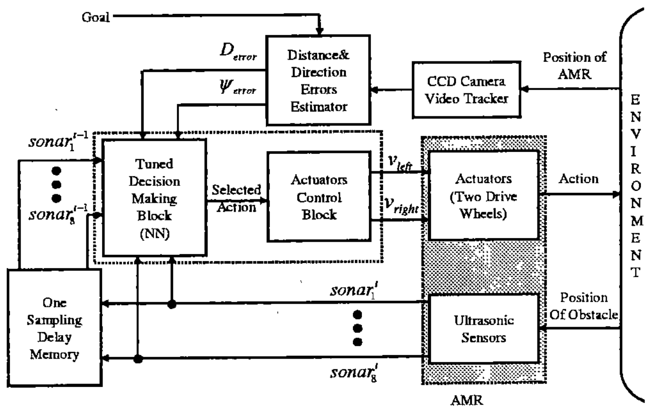


Fig. 3. Overall control system.

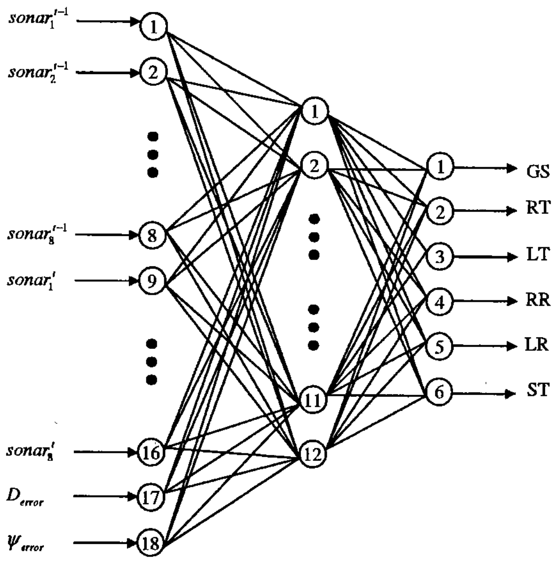


Fig. 4. Topology of neural network

layers, the neural network has one hidden layer with 12 neurons.

The sigmoid transfer functions of the hidden layer and the output layer are expressed by

$$F(x) = \begin{cases} \dfrac{2}{1+\exp(-x)} - 1 & \text{for hidden layer} \\ \dfrac{1}{1+\exp(-x)} & \text{for output layer} \end{cases} \qquad \cdots\cdots (4)$$

If the appropriate tuning parameters are obtained, the decision-making block can select the suitable action and realize the successful navigation of the AMR under unknown environments. However, it is difficult to obtain the suitable decision-making in advance. Therefore, the off-line tuning process of the decision-making block is required discussed in next section 4.

3.3 Actuator Control Block The tested AMR is controlled by the two independent driving wheels as shown in Fig.1. The objective of this block is to determine the speed of two driving wheels to realize the selected action from the decision-making block. In this research, the wheel speed is set to the discrete values expressed as follow.

- **GS** $(\nu_{left}, \nu_{right}) = (g_s.\nu_{max}, g_s.\nu_{max})$
- **RT** $(\nu_{left}, \nu_{right}) = (g_t.\nu_{max}, 0.05)$
- **LT** $(\nu_{left}, \nu_{right}) = (0.05, g_t.\nu_{max})$
- **RR** $(\nu_{left}, \nu_{right}) = (g_r.\nu_{max}, -g_r.\nu_{max})$
- **LR** $(\nu_{left}, \nu_{right}) = (-g_r.\nu_{max}, g_r.\nu_{max})$

where g_s, g_t, g_r are the speed gains for go straight action, turning action, and rotating action, respectively and are set between 0 to 1. ν_{max} is the maximum speed of tested robot and is set to 0.2 m/s.

Since stop action is to brake the robot, both speed for right and left driving wheels are set to 0. Therefore, stop speed gain is not needed.

The speed gains are incorporated into genotype and are simultaneously tuned with connection weights of NN by genetic algorithm.

4. Off-line Tuning of Decision-making Block and Actuator Control Block

In this section, the off-line tuning scheme of the decision-making block and actuator control block using the genetic algorithm is discussed. Fig.5 shows block diagram in the off-line tuning mode. To improve the precision in simulation, the functions of the ultrasonic sensors must be modeled accurately.

4.1 Simulator of Ultrasonic Sensors It is easy to calculate the distance and the direction of the obstacles geometrically in the simulation. However, the actual distance and direction of the obstacles are measured by ultrasonic sensors with wide directivity. Therefore, the tuning results based on the geometrical calculation may have the fetal defects due to the uncertain models of the ultrasonic sensors. To remove the model error, the simulator of the ultrasonic sensors is introduced.

The objective of the simulator is to simulate the actual outputs of the eight ultrasonic sensors mounted on the AMR by using the geometrically calculated distance and direction of the obstacles. The simulator consists of eight artificial neural networks ($2 \times 40 \times 10 \times 1$ neurons in each layer) corresponding to the sensors. The neural networks are adjusted by the back propagation algorithm in advance so that the output signal of each neural network corresponds to the actual output of the sensor. Fig.6 is the comparison between the outputs of the ultrasonic sensors and the tuned simulator, and demonstrates the validity of the tuned simulator of the ultrasonic sensors.

4.2 Danger-degree Estimator In the obstacle avoidance problem of AMRs, the concept of the danger-degree is effective and has been introduced by Maeda and Takegaki [12]. In the reference the static and dynamic danger-degrees are estimated by using the fixed fuzzy rules, then, the avoidance action is obtained by the decision table categorized by the danger-degrees.

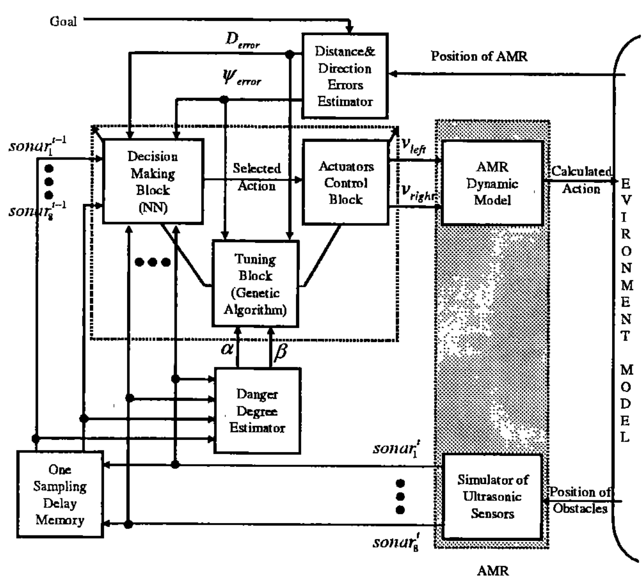


Fig. 5. Off-line tuning of decision-making block and actuator control block.

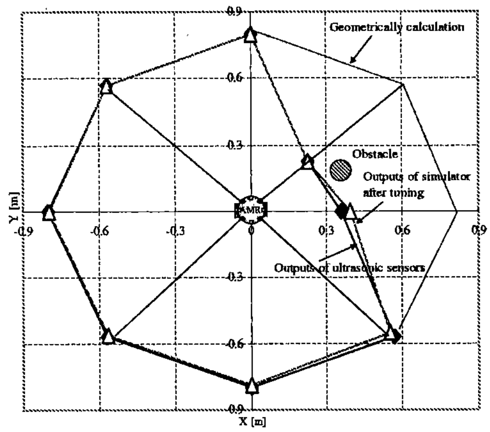


Fig. 6. Comparison between the outputs of ultrasonic sensor and the simulator.

In this paper, the static and dynamic danger-degrees for the obstacles are introduced. The static danger-degree focuses on the static relation between the AMR and the obstacles. Therefore, the outputs of the simulator are encoded as the inputs to static fuzzy estimator. On the other hand, the dynamic danger-degree represents the dynamic relation between robot and obstacles. The variations of the outputs of the simulator are encoded as the inputs to the dynamic fuzzy estimator. Based on the experience and knowledge of control engineers, the sets of the fuzzy rules are tuned beforehand.

The static and dynamic danger-degree can be expressed as

$$\alpha = \frac{\sum \mu_l \mu_\theta W_\alpha}{\sum \mu_l \mu_\theta} \quad \cdots\cdots (5)$$

$$\beta = \frac{\sum \mu_w \mu_\phi W_\beta}{\sum \mu_w \mu_\phi} \quad \cdots\cdots (6)$$

where μ_l, μ_θ are the grade values of distance and direction calculated by static danger-degree fuzzification membership functions, W_α is the weight predefined in static defuzzification singleton membership function. Similarly, μ_w, μ_ϕ are the grade values of distance and direction variations calculated by dynamic danger-degree fuzzification membership functions, W_β is the weight predefined in dynamic defuzzification singleton membership function.

4.3 Tuning Block

4.3.1 Coding The weight parameters of the neural network in the decision-making block and speed gains in actuator control block correspond to the genotype, and represent the organization of an individual i of a group of the population I. Then, the individual i is given by genetic codes as

$i : W^m_{0,0}, \ldots, W^m_{17,11}, W^n_{0,0}, \ldots, W^n_{11,5}, g_s, g_t, g_r$

Each individual is estimated at every trial and adjusted by the following selection, crossover and mutation operations.

4.3.2 Fitness Function Each individual is evaluated by the following fitness function

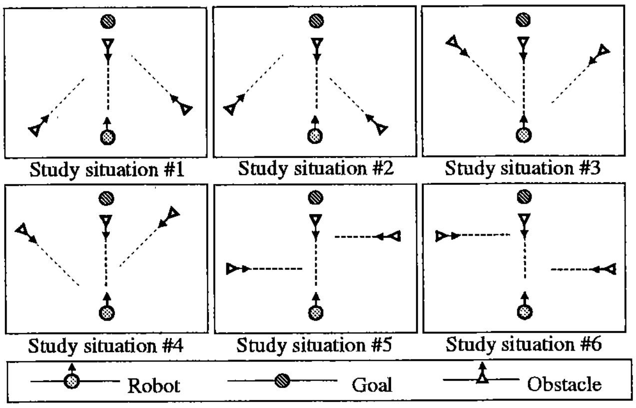


Fig. 7. Study situations for AMR.

$$f_{ave} = \frac{1}{6}\sum_{n=1}^{6} f_n \quad \cdots\cdots (7)$$

$$f_n = \frac{\left(e^{(n)}_\alpha + e^{(n)}_\beta + e^{(n)}_\Psi + e^{(n)}_d + e^{(n)}_c\right)}{T} + f^n_p \quad \cdots\cdots (8)$$

where $e^{(n)}_\alpha$ and $e^{(n)}_\beta$ are the sums of static and the dynamic danger-degrees, while $e^{(n)}_d$ and $e^{(n)}_\Psi$ are the sums of the distance and the direction errors with respect to goal, $e^{(n)}_c$ is the sum of energy consumption, at study situation n. f_p is the penalty. T is the real time from the starting point to the goal at every trial, and n corresponds to the study situation shown in Fig.7.

The sums of the static and dynamic danger-degrees are given

$$e_\alpha = \frac{\sum_{t=0}^{T}\left(\sum_{j=1}^{3}|\alpha_j(t)| + |\alpha_7(t)| + |\alpha_8(t)|\right)}{5} \quad \cdots\cdots (9)$$

$$e_\beta = \frac{\sum_{t=0}^{T}\sum_{j=1}^{8}|\beta_j(t)|}{8} \quad \cdots\cdots (10)$$

where α_j and β_j are the static and dynamic danger-degrees discussed in the above section. According to the define of static danger-degree, the obstacles in backside are not considered. Therefore, e_α doesn't include informations from the ultrasonic sensors US4, US5 and US6.

The sums of the distance and direction errors are given

$$e_\Psi = k_\Psi \sum_{t=0}^{T} \Psi_{error}(t) \quad \cdots\cdots (11)$$

$$e_d = k_d \sum_{t=0}^{T} d_{error}(t) \quad \cdots\cdots (12)$$

where d_{error} and Ψ_{error} are the distance and the direction errors given by Equation (1) and (2), k_Ψ and k_d are the gains and are set to 0.15.

The sum of energy consumption is given by

$$e_c = \sum_{t=0}^{T} \begin{cases} g_s & \text{for GS action} \\ g_t & \text{for RT and LT actions} \\ g_r & \text{for RR and LR actions} \\ 0 & \text{for ST action} \end{cases} \quad \cdots (13)$$

In equation (13), it's assumed that the consuming energy is proportional to the wheel speed. Although the higher speed motion consume much energy, the traveling period, T, is shorted.

Furthermore, the penalty f_p is given by

$$f_p = \begin{cases} 500 & \text{not arrive to the goal} \\ 0 & \text{arrive to the goal} \end{cases} \quad \cdots\cdots (14)$$

Equation (14) implies that the large penalty is added if and only if the AMR can not arrive at the goal within the given period T_n.

4.3.3 Genetic Operations Fig.8 shows the outline of the proposed genetic operations. The number of individuals in each generation is set to 100. Firstly, the individuals are applied to the navigation of the AMR and evaluated by the fitness function in order. As a result of estimation, the individuals are ranked. The best individual is copied and remains in the next generation without modification. On the other hand, the 10 worst individuals are selected and can not remain the next generation. After ranking, the rest are randomly created by combining two individuals based on crossover strategy. Some individuals expect for the best individual are modified by mutation with probability of 0.3. Consequently, the individuals in the next generation are obtained by the genetic operations.

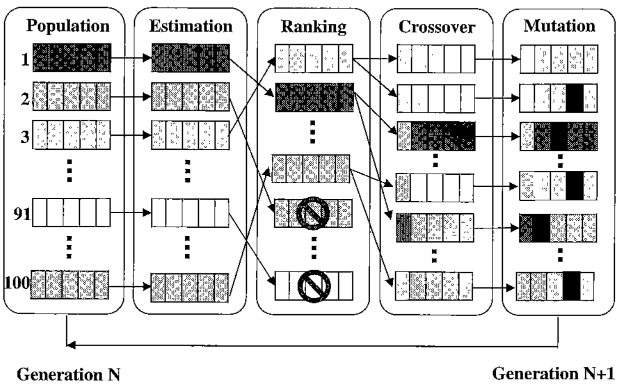


Fig. 8. Genetic operations.

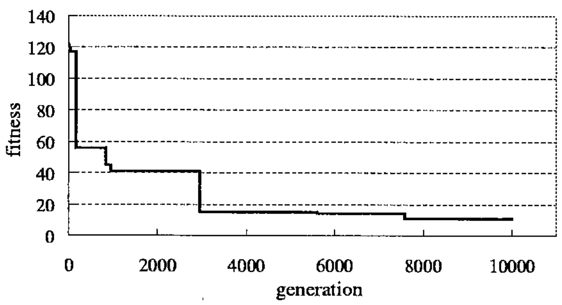


Fig. 9. Estimated value of best individual in each generation.

4.4 Tuning Results To tune the decision-making neural network and speed gains, the navigations of the AMR in the various situations shown in Fig.7 are simulated.

Fig.9 shows the estimated value of the best individual in each generation. The estimated value decreases as the generation processes. This simulation result implies that the suitable decision-making neural network and speed gains are tuned by the genetic algorithm.

5. Simulation and Experimental Results

In order to confirm the validity of the proposed navigation system, both the simulation and real world experiments of AMR are conducted.

5.1 Simulation Result Fig10 shows the simulation result under the condition of 3 obstacles. The obstacles go straight with a constant speed of 0.15 m/s.

The AMR can arrive to the goal with avoiding three coming obstacles. This result demonstrates that the reasonable navigation is obtained by the proposed scheme.

5.2 Experimental Results

5.2.1 Navigation under Static Environments Three kinds of classical static situations are used in the experiments.

Although we did not use these situations during offline tuning of decision-making block, the AMR can navigate to the goal successfully without hitting the obstacles.

5.2.2 Navigation under Dynamic Environments Two kinds of dynamic environment are prepared in the experiments, with one dynamic obstacle and two dynamic obstacles.

Fig.14 and Fig.15 show the trajectories of the AMR

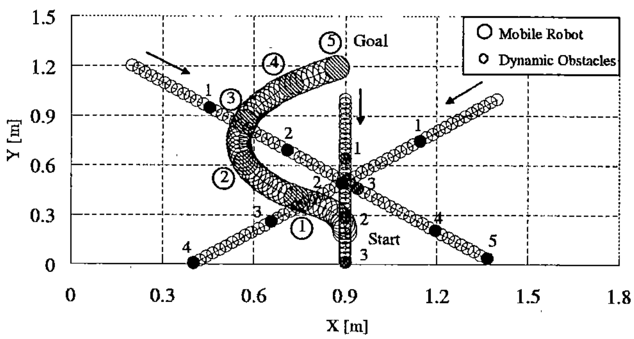


Fig. 10. Trajectory under condition with 3 dynamic obstacles.

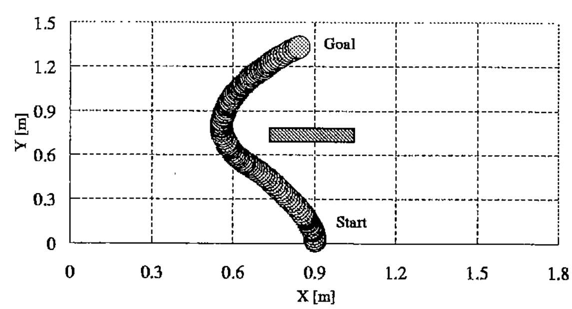


Fig. 11. Trajectory under static panel condition.

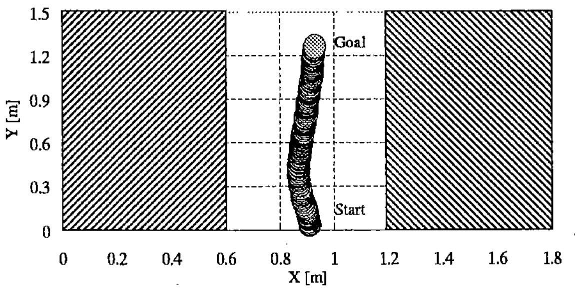


Fig. 12. Trajectory under corridor condition.

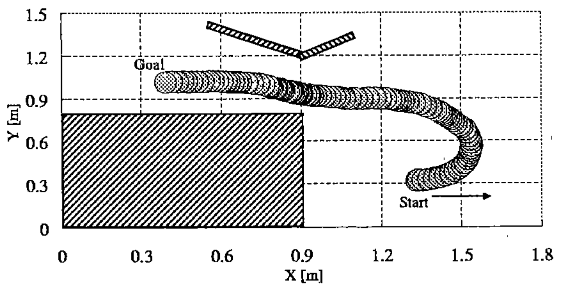


Fig. 13. Trajectory under unstructured condition.

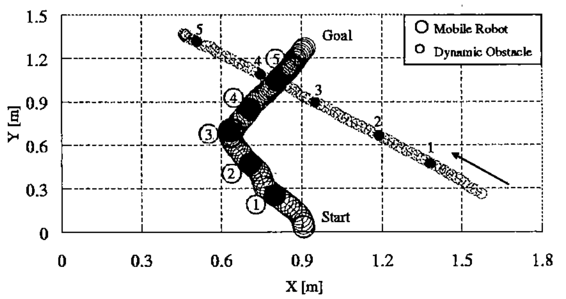


Fig. 14. Trajectory under condition with one dynamic obstacle.

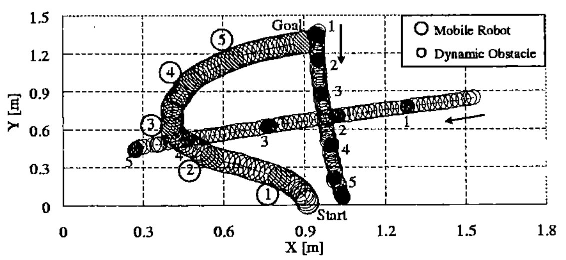


Fig. 15. Trajectory under condition with two dynamic obstacles.

and obstacles under two different dynamic situations. Fig.16 shows the selected action of decision-making neural network, and Fig.17 depicts the estimate parameters during navigation under condition in Fig.14. It can be seen that both danger-degrees can be properly evaluated by the fixed fuzzy rules and the decision-making block with the optimal weights set can select the reasonable action to avoid obstacles on the way to the goal.

6. Conclusion

In this paper, a navigation problem for an autonomous

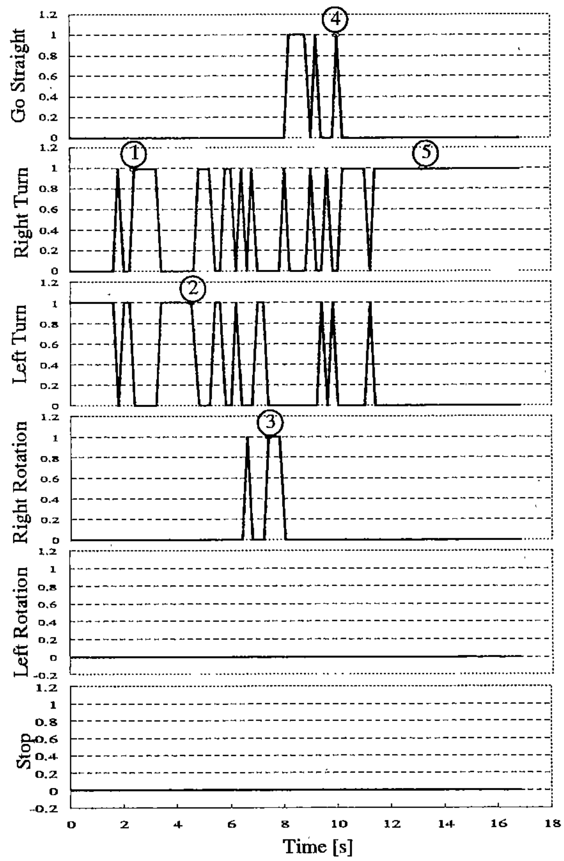


Fig. 16. Actions selected by Decision-making neural network under condition in Fig.14.

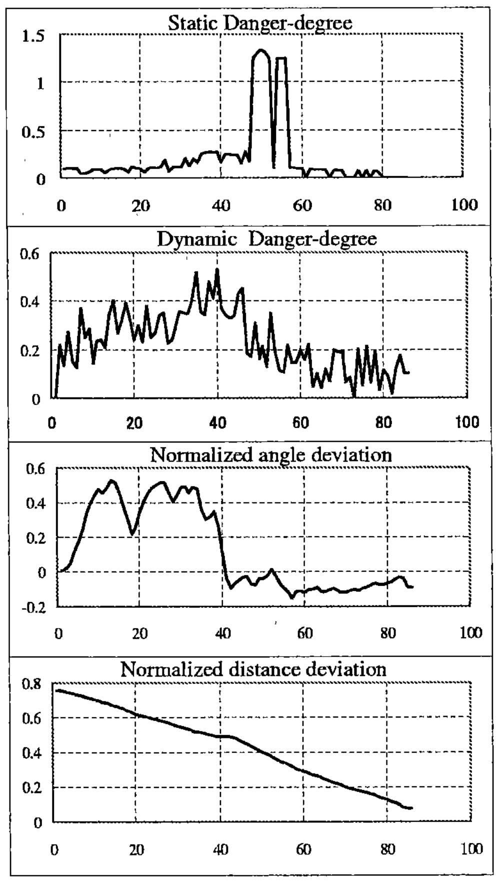


Fig. 17. Estimate parameters under condition in Fig.14.

mobile robot under unknown environment is considered. To generate the suitable avoidance action for obstacle and navigate to the goal successfully, the decision-making strategy tuned by genetic algorithm is proposed in this research. The static and dynamic danger-degrees, the distance and direction errors and the energy consumption are used to estimate the set of weight parameters of decision-making block and speed gains in actuator control block. After off-line tuning, we implemented the control system in the tested mobile robot. In our experiments, the mobile robot was navigated successfully in various environments. Experimental results imply that proposed control system is correct and reasonable and has adaptability in the real environment. The future work will extend the scheme to obtain more competencies for AMR, meanwhile vision system will be mounted on robot to improve the efficiency of target recognition and obstacle avoidance. Finally, this work was supported by MEXT, KAKENHI (14750372).

(Manuscript received Nov. 25, 2002,
revised March 7, 2003)

References

(1) T.K. Win: "Collision avoidance control for autonomous mobile robot based on operator's skill", Master Thesis, University of Tokushima, (2000)

(2) R. Brooks: "A robust layered control system for a mobile robot", *Robotics and Automation, IEEE Journal of [legacy, pre-1998]*, Vol.2, Issue: 1, pp.14–23 (1986)

(3) S. Kristensen: "Sensor planning with bayesian decision theory", Laboratory of Image Analysis, Aalborg University, Technical Report (1995)

(4) Y. Cang and W. Danwei: "A novel navigation method for autonomous mobile vehicles", *Journal of Intelligent and Robotic Systems*, Vol.32, pp.361–388 (2001)

(5) D.E. Rumelhart, G.E. Hinton, and R.J. Williams: "Learning internal representation by error propagation", DE Rumelhart, JL McClelland and the PDP Research Group (eds), Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol.1, chapter 8, Bradford Books (MIT Press), Cambridge, MA (1986)

(6) A.G. Barto, S.J. Bradtke, and S.P. Singh: "Learning to act using real-time dynamic programming", *Artificial Intelligence, Speical Volume: Computational Research on International and Agency*, 72, pp.81–138 (1995)

(7) L. Booker, D.E. Godberg, and J.H. Holland: "Classifier systems and genetic algorithms", *Artificial Intelligence*, 40, pp.235–282 (1989)

(8) T. Kohonen: "Self-organized formation of topologically correct feature maps", *Biological Cybernetics*, 43, pp.59–69 (1982)

(9) D.T. Cliff, I. Harvey, and P. Husbands: "Explorations in evolutionary robotics", *Adaptive Behavior*, 2, pp.73–110 (1993)

(10) S. Nolfi and D. Floreano: "Learning and evolution", *Autonomous Robots*, 7(1), pp.89–113 (1999)

(11) R. Ashwin, A. Ronald, B. Gary, and P. Michael: "Using genetic algorithm to learn reactive control parameters for autonomous robot navigation", *Adaptive Behavior*, Vol.2, Issue 3, pp.277–304 (1994)

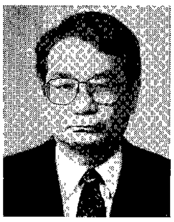
(12) Y. Maeda and M. Takegaki: "Collision avoidance control among moving obstacles for a mobile robot on the fuzzy reasoning", *Journal of the Robotics Society of Japan*, Vol.6, No.6, pp.518–522 (1988)

Fei Wang (Student Member) received the B.E. and M.E. degrees in Vehicle School at Harbin Institute of Technology, China, in 1997 and 1999. He is currently working toward the D.Eng. degree in system engineering at The University of Tokushima.

Takuya Kamano (Member) received his B.E. degree in electrical engineering from the University of Tokushima in 1980, and his M.E. degree in electrical engineering from kyushu University in 1983. He received his D.Eng. degree in electrical engineering from kyoto University in 1987. He has been an associate professor in the Department of Electrical and Electronic Engineering at the University of Tokushima since 1988. He is a member of SICE, ISCIE, the Japan Society of Mechanical Engineers, SOFT, RSJ, and IEEJ.

Takashi Yasuno (Member) received his B.E degree in electrical engineering, his M.E. degree in electrical and electronic engineering, and his D.Eng. degree in system engineering from the University of Tokushima in 1991, 1993, 1998. He has been an assistant professor in the Department of Electrical and Electronic Engineering at the University of Tokushima since 2001. He is a member of SICE, the institute of Systems, ISCIE, the Japan Society of Applied Electromagnetics and Mechanics, SOFT, RSJ, and IEEJ.

Takayuki Suzuki (Member) received his B.E. and M.E. degrees in electrical engineering from the University of Tokushima in 1962 and 1966. From 1962 to 1968, he was a research associate and an assistant professor in the Department of Electrical Engineering at the University of Tokushima, respectively. He has been a professor in the Department of Electrical and Electronic Engineering at the University of Tokushima since 1982. He received his D.Eng. degree in electrical engineering from kyoto University in 1975. He is a member of SICE, ISCIE, the Japan Association for Wind Engineering, and IEEJ.



Hironobu Harada (Member) received his B.E. degree in electrical engineering and mechanical engineering from the University of Tokushima in 1970 and 1972. He is a technical assistant in the Department of Electrical and Electronic Engineering at the University of Tokushima. He is a member of SICE and IEEJ.