The Sixteenth International Symposium on Artificial Life and Robotics 2011 (AROB 16th '11),
B-Con Plaza, Beppu,Oita, Japan, January 27-29, 2011

Angle-based neuro-fuzzy navigation for autonomous mobile robots

Shu Hosokawa, Joji Kato, Kazushi Nakano, Kazunori Sakurama

Dept. of Electronic Eng.,The University of Electro-Communications, Japan
(*Tel : 81-42-443-5190; Fax : 81-42-443-5183*)
(*hosokawa@francis.ee.uec.ac.jp*)

Abstract: This paper presents a neuro-fuzzy navigation method for mobile robots based on local sensors mounted on the robot. This method is not sensitive to sensor noise, and is able to automatically adjust the internal parameters given by the teacher signal. However, most of previous studies dealt with two-wheel driven robots have focused on the acceleration control of their wheels. For this reason, it is difficult to generate teacher signals for robots with many actuators such as omnidirectional mobile robots. In this paper, we propose a method of neuro-fuzzy navigation based on control of rotation speed of the robots. We demonstrate the validity of our proposed method through simulations and experiments.

Keywords: Omnidirectional mobile robot, Neuro fuzzy, Robot navigation, Local sensor

I. Introduction

Recently, the autonomous mobile robots working in dangerous environments such as disaster sites have been studied. These robots cannot get information about their operating environments in advance. For this reason, these robots require navigation methods to a target point based on information from local sensors mounted on the robots.

There are some methods based on the fuzzy inference techniques using local sensors on the robots to navigate to a target point [1][2]. Those methods can reduce the effect of observation noises from local sensors, but adjustments of the parameters by experts are necessary. Automatic parameter turning methods which incorporate neural network structures into fuzzy inference systems have been proposed in [3][4]. These controllers are able to adjust the parameters automatically by giving desired moving patterns. However, most of previous studies deal with the two wheel robots considering the acceleration of the motors as input. But when applying these methods to more complex systems (for example: snake-like robots, omnidirectional mobile robots and so on), the complexity makes it difficult to generate teacher signals.

In this paper, we propose a method using the rotation speeds of the robots as the input for the robot. By using the rotation speed instead of the acceleration of each actuator, we can generate the teacher signals for the controller. The validity of the proposed method is demonstrated through simulations and experiments.

II. Omnidirectional mobile robot

Figure 1 shows an omnidirectional mobile robot in this study, where θ is the angle between y-axis and a wheel, V_x is the velocity at the front direction of the

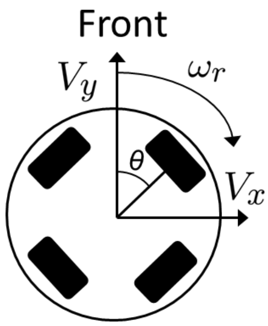

Figure 1 Omnidirectional mobile robot

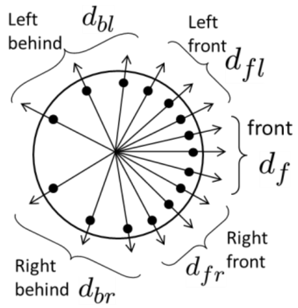

Figure 2 Obstacle sensor placements

robot, V_y is the velocity at the lateral direction of the robot, and w_r is the velocity of the rotation of the robot. In this study, the rotation speed of each wheel is controlled by a suitable controller (for example: PID), and is assumed to accurately rotate according to the reference value. The following equation shows the relationship between the moving velocity of the robot and the velocity of each wheel:

$$\begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \end{bmatrix} = \frac{1}{R_w} \begin{bmatrix} \sin(\theta) & -\cos(\theta) & R \\ \sin(\theta) & \cos(\theta) & R \\ -\sin(\theta) & \cos(\theta) & R \\ -\sin(\theta) & -\cos(\theta) & R \end{bmatrix} \begin{bmatrix} V_x \\ V_y \\ w_r \end{bmatrix} \tag{1}$$

where R is the radius of the robot, and R_w is the radius of the wheel. In this study, we design a controller using two pieces of information; the distance between the obstacle and the robots, the angle of the target point from the front direction of the robot.

Figure 2 shows the arrangement of fifty obstacle detection sensors mounted on the robot. Those sensors are divided into five groups of three adjacent sensors {right behind, right front, front, left front, left behind}. The output value of each sensor group is selected to take a smallest value in the groups. The angle of the target direction is given from the front of the robot. In this paper, we assume a constant velocity at the lateral direction for simplicity.

III. Angle-based neuro-fuzzy navigation

This section, explains the proposed method of angle-based neuro-fuzzy controller. At first, we explain the controller structures and how to adjust the parameters of the fuzzy membership functions.

1. Controller structure

Figure 3 shows the structure of the proposed angle-based neuro-fuzzy controller. This controller consists of a five-layer neural network. The first layer receives the target direction and distance information from the obstacle sensors. The second layer converts an input value to a fuzzy grade using the fuzzy inference logic. Membership functions of the target direction consist of five fuzzy sets {right behind, right front, front, left front, left behind}, in the form of triangular type functions:

$$p_{ij} = \begin{cases} 1 - \frac{2|u_i - m_{ij}|}{\sigma_{ij}}, m_{ij} - \frac{\sigma_{ij}}{2} < u_i < m_{ij} + \frac{\sigma_{ij}}{2} \\ 0.001, \text{ otherwise} \end{cases} \tag{2}$$

where i is a input number, j is a fuzzy set number, p_{ij} is the fuzzy grade, m_{ij} is the center value of the membership function, u_i is the input value, σ_{ij} is the width of the membership function. Otherwise, the membership function of the obstacle sensor consists of two fuzzy sets to determine the distance from the obstacle. A Z-type function in (3) is adopted as the membership function that represents the distance between the robot and the obstacle:

$$p_{ij} = \begin{cases} 0.001, u_i \geq m_{ij} + \frac{\sigma_u}{2} \\ 1, u_i < m_{ij} \\ 1 - \frac{2|u_i - m_{ij}|}{\sigma_{ij}}, \text{otherwise} \end{cases} \tag{3}$$

while, a S-type function in (4) is adopted as the membership function that represents the closeness between the robot and the obstacles.

$$p_{ij} = \begin{cases} 0.001, u_i \leq m_{ij} - \frac{\sigma_u}{2} \\ 1, u_i > m_{ij} \\ 1 - \frac{2|u_i - m_{ij}|}{\sigma_{ij}}, \text{otherwise} \end{cases} \tag{4}$$

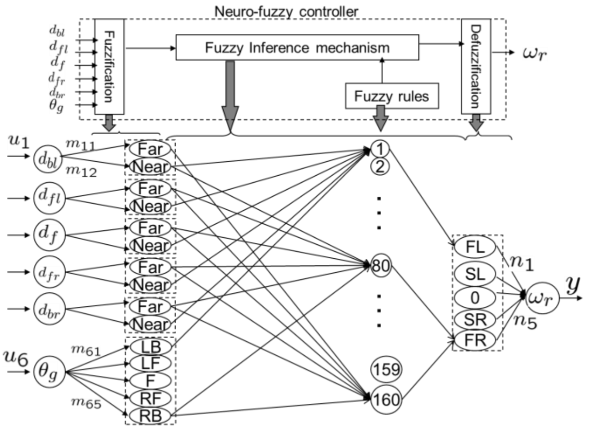

Figure 3 Angle-based neuro-fuzzy controller

The third layer represents the fuzzy controller based on the if-then rules, which has the connection with the second layer as the antecedent, and the connection with the fourth layer as the consequent. As for the if-then rule, there are 160 rules from five fuzzy sets of the target direction and two fuzzy sets of five groups of obstacle sensors. The fourth layer represents the output linguistic grades. The fifth layer is the output of the controller, which gives the center of gravity defuzzification of fuzzy sets of the four layers. The center of gravity is calculated using the lowest grade in the antecedent of each fuzzy rule:

$$y = \frac{\sum_{k=1}^{160} v_k q_k}{\sum_{k=1}^{160} q_k} \tag{5}$$

$$q_k = \min\{p_1 k_1, p_2 k_2, p_3 k_3, p_4 k_4, p_5 k_5, p_6 k_6\} \tag{6}$$

where y is the output value of the rotation value, k is the number of fuzzy rule sets, v_k is the center value of the fuzzy membership function, and q_k is the fitness of the k-th fuzzy rules.

2. Learning algorithm

We explain a parameter adjustment law adopted by our proposed method to use teacher signals. The proposed method has the parameters; the center value of fuzzy membership function and the width of fuzzy membership function. In this study, we update only the center values of fuzzy membership functions.

Adjustment of the parameters is performed by using the evaluation function described by the following equation:

$$E = \frac{1}{2}(y - \hat{y})^2 \tag{7}$$

where E is the error, $\hat{y}$ is the teacher signal. This evaluation is based on the least mean square method. $\boldsymbol{Z}$ as the median to be updated, the update formula is expressed as

$$\boldsymbol{Z}(t+1) = \boldsymbol{Z}(t) - \varepsilon \frac{\partial E}{\partial \boldsymbol{Z}} \tag{8}$$

where ε is the learning rate, $\boldsymbol{Z}(t)$ is the parameter vector at the update number t. Each parameter is updated by the following equation:

$$m_{ij}(t+1) = m_{ij}(t) - \varepsilon \frac{\partial E}{\partial m_{ij}} \tag{9}$$

$$n_s(t+1) = n_s(t) - \varepsilon \frac{\partial E}{\partial n_s} \tag{10}$$

where, if $q_k = p_{ij}$ and $p_{ij} \neq \text{const}$,

$$\frac{\partial E}{\partial m_{ij}} = -2\left\{(y-\hat{y}) \frac{v_k \sum_{k=1}^{160} q_k - \sum_{k=1}^{160} v_k q_k}{\left(\sum_{k=1}^{160} q_k\right)^2}\right\} \times \frac{\operatorname{sign}\left(u_i - m_{ij}\right)}{\sigma_{ij}}$$

otherwise

$$\frac{\partial E}{\partial m_{ij}} = 0 \quad .$$

.
And also, if $v_k = n_s$,

$$\frac{\partial E}{\partial n_s} = (y-\hat{y}) \frac{q_k}{\sum_{k=1}^{160} q_k}$$

otherwise

$$\frac{\partial E}{\partial n_s} = 0$$

Teacher signals are used in expressions to produce more updates from the history of the human-robot operation. The teacher signals has the three types of parameters; the angle of the target point from the front direction of the robot, the distance from the obstacles were measured by the local sensor mounted on the robot, and the velocity of the rotation of the robot were controlled by a human. Parameter adjustment is repeated until the error falls below the set value δ $(0 < \delta)$.

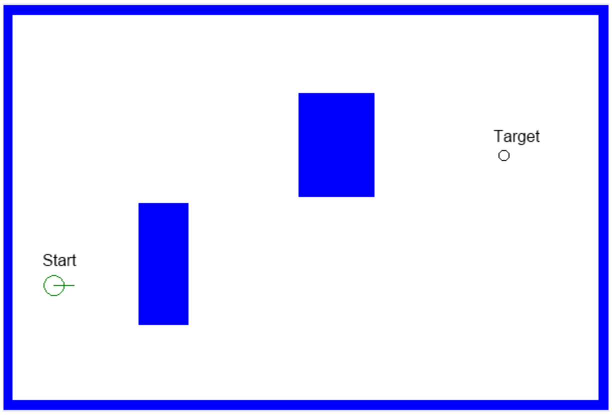

Figure 4 Training environment

Table 1 Simulation parameters

Robot radius	10 cm
Working space	405 cm × 605 cm
Measuring range of the obstacle sensor	0 cm ~ 100 cm
Translation velocity	20 cm/s
Control cycle	200 ms

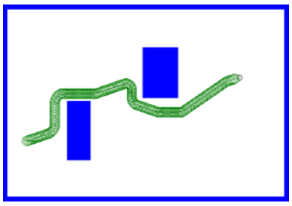

Figure 5 Trajectory for training data

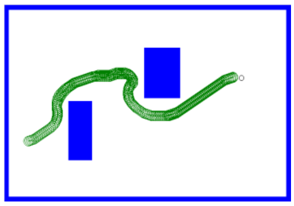

Figure 6 Trajectory after parameter adjustment

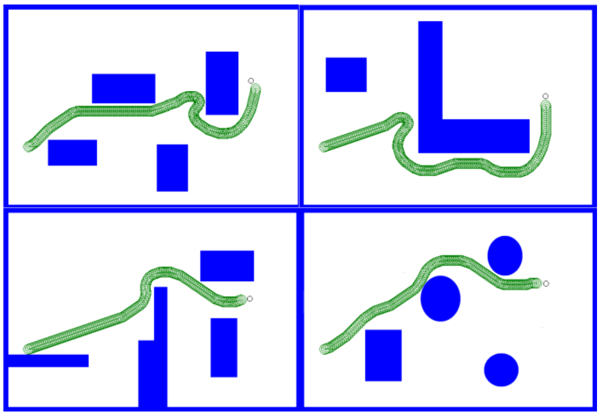

Figure 5 Results in various environments

IV. Simulation and experimental results

This section shows simulation and experiment results of our proposed method. First, we explain how to generate the teacher signal in the training environment, and adjust the parameters of fuzzy membership functions. Next, we show some different environments. Finally, we demonstrate the results of the actual robot.

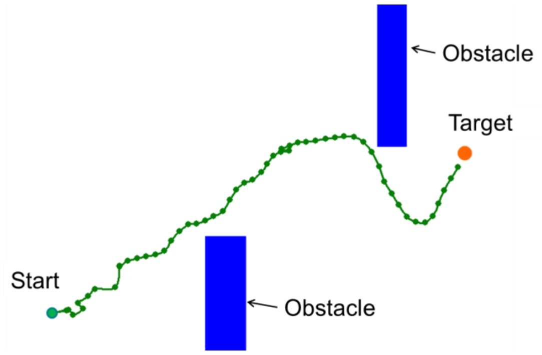

Figure 6 Actual results

Figure 4 shows the environment in the simulation. Table 1 is the list of the parameters used in these simulations. The simulation purpose is to move a robot to a goal position on the left side of each environment from the start at the right side without colliding against obstacles.

Figure 5 shows the trajectory of the robot operated by a human teacher. Figure 6 shows the trajectories of the robot after the parameter adjustment based on the teacher. By adjusting the parameters of fuzzy membership functions, we can see that the robot reaches the target without colliding against obstacles.

Next, we show the result in some different environments with pre-adjusted parameters in the previous environment. The starting position and the target point are the same as the previous environment. Figure 7 shows the results in the four simulation environments in different shapes, where the numbers of obstacles are different. We can see that the robot reach the target in all the environments without re-adjusting the parameters.

Finally, we apply our proposed method to the actual robot. We prepare a simulated environment created by the teacher signals shown in Figure 5. The parameters of membership functions are obtained by the simulation. Figure 8 shows the trajectory obtained by the actual robot. We can see that the actual robot reach the target without colliding with obstacles.

VI. Conclusion

In this study, we proposed an angle-based neuro-fuzzy navigation method. We control the rotation speed of robots instead of the acceleration of each motor. Thus, a controller structure was too compact to simplify the generation of teaching signals even when applying the robot to a complex structure. We showed the validity of our proposed method through simulations and experiments.

REFERENCES

[1] Eugenio Aguirre and Antonio Gonzalez (2003), A fuzzy perceptual model for ultrasound sensors applied to intelligent navigation for mobile robots. Applied Intelligence 19, 171-187

[2] S.M.Raguraman, D.Tamilselvi and N. Shivvakumar (2009), Mobile robot navigation using fuzzy logic controller. International conference on control , automation, communication and energy conservation

[3] Kai-Tai-Song and Liang-Hwang Sheen (2000), Heuristic fuzzy-neuro network and its application to reactive navigation of a mobile robot. Fuzzy Set and Systems 110, 331-1340

[4] Anmin Zhu and Simon X. Yang (2007), Neurofuzzy-based approach to mobile robot navigation in unknown environments. IEEE trans. on Systems, Man, and Cybernetic – Part C; Applications and Review 37(4), 610-621