2004년도 대한전기학회 하계학술대회 논문집 2004.7.14-16

이동 로봇의 경로 추종을 위한 웨이블릿 퍼지 신경 회로망 기반 직접 적응 제어 시스템

°오준섭*, 박진배*, 최윤호**
*연세대학교 전기전자공학과, **경기대학교 전자공학부

Direct Adaptive Control System for Path Tracking of Mobile Robot Based on Wavelet Fuzzy Neural Network

Joon Seop Oh*, Jin Bae Park*, Yoon Ho Choi**
*Dept. of Electrical & Computer Eng., Yonsei Univ., **School of Electronic Eng., Kyonggi Univ.

Abstract - In this paper, we present a novel approach for the structure of Fuzzy Neural Network(FNN) based on wavelet function and apply this network structure to the solution of the tracking problem for mobile robots. Generally, the wavelet fuzzy model(WFM) has the advantage of the wavelet transform by constituting fuzzy basis function(FBF) and the conclusion part to equalize the linear combination of FBF with the linear combination of wavelet functions. However, it is very difficult to identify the fuzzy rules and to tune the membership functions of the fuzzy reasoning mechanism. Neural networks, on the other hand, utilize their learning capability for automatic identification and tuning. Therefore, we design a wavelet based FNN structure(WFNN) that merges these advantages of neural network, fuzzy model and wavelet. To verify the efficiency of our network structure, we evaluate the tracking performance for mobile robot and compare it with those of the FNN and the WFM.

1. Introduction

Motion control of mobile robots is a typical nonlinear tracking control issue and has been discussed with different control schemes such as PID, GPC, sliding mode, predictive control etc[1]-[3]. Intelligent control techniques, based on neural networks and fuzzy logic, have also been developed for path tracking control. Even though these intelligent control strategies have shown effectiveness, especially for nonlinear systems, they have certain drawbacks derived from their own characteristics. While conventional neural networks have good ability of self-learning, they also have some limitations such as slow convergence, the difficulty in reaching the global minima in the parameter space, and sometimes instability as well[4]-[5]. In case of fuzzy logic, it is a human-imitating logic, but lacks the ability of self-learning and self-tuning. Therefore, in the research on the intelligent control, FNNs are devised to overcome these limitations and to combine the advantages of both neural networks and fuzzy logic[6]-[7]. This provides a strong motivation for using FNNs for modeling and controlling nonlinear systems. And the wavelet fuzzy model(WFM) has the advantage of the wavelet transform by constituting FBF, the conclusion part to equalize the linear combination of FBF with the linear combination of wavelet functions and modifying fuzzy model to be equivalent to wavelet transform. The conventional fuzzy model cannot give the satisfactory result to the transient signal. On the contrary, in the wavelet fuzzy model, the accurate fuzzy model can be obtained because the energy compaction by the unconditional basis and the description of a transient signal by wavelet basis functions are distinguished. Therefore, we design a FNN structure based on wavelet that merges these advantages of neural network, fuzzy model and wavelet. And we present the direct adaptive control system using WFNN structure. The control signals are directly obtained by minimizing the difference between the reference track and the pose of a mobile robot that is controlled through a WFNN. The control process is a dynamic on-line process that uses the WFNN trained by GD method. Through computer simulations, we demonstrate the effectiveness and the feasibility of the proposed control method.

2. Wavelet Based Fuzzy Neural Network

The basic idea of WFNN is to realize the process of fuzzy reasoning of wavelet fuzzy model by the structure of a neural network and to make the parameters of fuzzy reasoning be expressed by the connection weights of a neural network. WFNNs can automatically identify the fuzzy rules by modifying the connection weights of the networks using the gradient descent scheme. Among various fuzzy inference methods, WFNNs use the sum-product composition. The functions that are implemented by the networks must be differentiable in order to apply the gradient descent scheme to their learning.

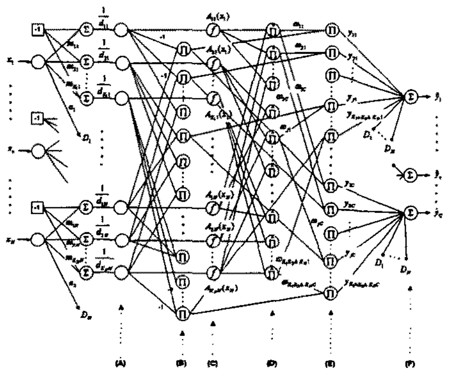


Fig. 1 WFNN Structure

Fig. 1 shows the configuration of WFNN, which has N inputs $(x_1, x_2, \cdots, x_N)$, and C output $(y_1, y_2, \cdots, y_c)$, and K_n membership functions in each input x_n. WFNN can be divided into two parts according to the process of the fuzzy reasoning: the premise part and the consequence parts. The premise part consists of nodes (A), (C), (D) and the consequence parts consist of nodes (D) through (F). The nodes (B) and (E) are used to equalize the linear combination of FBF with the linear combination of wavelet functions for the advantage of wavelet transform by constituting FBF and conclusion part. Therefore, the output node (F) is equivalent to wavelet transform. Consequently, in our WFNN structure, the output y_c is calculated as follows:

$$\hat{y}_c = \sum_{n=1}^{N} a_{nc}x_n + \sum_{j=1}^{R} B_{jc}\Phi_j \tag{1}$$

where,

$$B_{jc}\Phi_j = w_{jc}\frac{\prod_{n=1}^{N}\left(-\left(\frac{x_n - m_{k_n n}}{d_{k_n n}}\right)\right)exp\left(-\frac{1}{2}\left(\frac{x_n - m_{k_n n}}{d_{k_n n}}\right)^2\right)}{\sum_{j=1}^{R}\prod_{n=1}^{N}\left(exp\left(-\frac{1}{2}\left(\frac{x_n - m_{k_n n}}{d_{k_n n}}\right)^2\right)\right)},$$

Mother wavelet: $\phi(z) = -z exp(-\frac{1}{2}z^2)$, $z = \frac{x - m}{d}$

k_n: k-th fuzzy variable of n-th input, N: input Num., K_n: fuzzy variable Num of input n, R: wavelet Num.

The consequence parts consist of nodes (D) through (F) and the fuzzy reasoning is realized as:

R^j : *If* x_1 *is* $A_{k_1 1}, \cdots x_n$ *is* $A_{k_n n} \cdots$ and x_N *is* $A_{k_N N}$

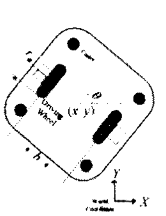
then $y_{jc} = w_{jc}$ ($j = 1, 2, \cdots, R$ and $c = 1, 2, \cdots, C$)

where, R^j is the j-th fuzzy rule, $A_{k_n n}$ is fuzzy variable in the premise, w_{jc} is a constant. The weights w_{jc} are modified to identify fuzzy rules using the gradient descent method.

3. Path tracking control for mobile robot using WFNN

3.1 Dynamic Model of Mobile Robot

The robot used in this paper is composed of two driving wheels and four casters and is fully described by a three dimensional vector of generalized coordinates constituted by the coordinates of the midpoint between the two driving wheels, and by the orientation angle with respect to a fixed frame as shown in Fig 2. We have the equation for motion dynamics as the follows:



$$\begin{bmatrix} X_{k+1} \\ Y_{k+1} \\ \theta_{k+1} \end{bmatrix} = \begin{bmatrix} X_k \\ Y_k \\ \theta_k \end{bmatrix} + \begin{bmatrix} \delta d_k cos(\theta_k + \frac{\delta\theta_k}{2}) \\ \delta d_k sin(\theta_k + \frac{\delta\theta_k}{2}) \\ \delta\theta_k \end{bmatrix}$$

$$\delta d = \frac{d_r - d_l}{2} \quad d\theta = \frac{d_r - d_l}{b} \tag{2}$$

Fig. 2 Mobile Robot Model

where, δd and $\delta\theta$ are linear velocity and angular velocity. And d_r, d_l and b are two incremental distances of two driving wheels and distance between these two wheels, respectively.

3.2 The direct adaptive control system using WFNN

The purpose of our control system is to minimize the state errors $E(e_x, e_y, e_\theta)$ between reference trajectory and controlled trajectory of a mobile robot. The overall control system is shown in Fig. 3. A WFNN controller calculates the control input $u_k = [\delta d_k, \delta\theta_k]^T$ by training the inverse dynamics of plant iteratively. But, the updating of WFNN parameters through the variation rate $J(\gamma, Y)$ in the gradient-descent method cannot be calculated directly. So, we train the parameters of WFNN through the transformation of the output error of plant. In this structure, inputs are composed of errors between reference trajectory and controlled trajectory, and outputs are control variables. Each control variable is as shown in Eq. (3).

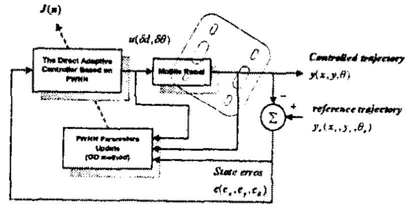


Fig. 3 Direct adaptive control system

$$\begin{aligned} \delta d_k &= \sum_{n=1}^{3} a_{nd}e_n + \sum_{j=1}^{R}\hat{y}_{jd} = \sum_{n=1}^{3} a_{nd}e_n + \sum_{j=1}^{R} B_{jd}\Phi_j \\ \delta\theta_k &= \sum_{n=1}^{3} a_{n\theta}e_n + \sum_{j=1}^{R}\hat{y}_{j\theta} = \sum_{n=1}^{3} a_{n\theta}e_n + \sum_{j=1}^{R} B_{j\theta}\Phi_j \end{aligned} \tag{3}$$

Training Procedure :

· Definition of the following cost function so as to train the WFNN controller based on direct adaptive control technique.

$$C = \frac{1}{2}\{(x_r - x)^2 + (y_r - y)^2 + (\theta_r - \theta)^2\} \tag{4}$$

· Calculation of the partial derivative of the cost function with respect to the parameter set of a WFNN controller.

$$\begin{aligned} \frac{\partial C}{\partial \gamma_c} &= -e_x\frac{\partial x}{\partial \gamma_c} - e_y\frac{\partial y}{\partial \gamma_c} - e_\theta\frac{\partial \theta}{\partial \gamma_c} \\ &= -e_\theta\frac{\partial x}{\partial u}\frac{\partial u}{\partial \gamma_c} \cdots\cdots = -E_Y J(u)\frac{\partial u}{\partial \gamma_c} \end{aligned} \tag{5}$$

where, $e_x = x_r - x$, $e_y = y_r - y$, $e_\theta = \theta_r - \theta$ and $J(u) = \frac{\partial Y}{\partial u}$ is the feedforward Jacobian of plant and is as follows.

$$J(u) \quad \begin{bmatrix} cos(\theta_k + \frac{\delta\theta_k}{2}) & -\frac{\delta d_k}{2}sin(\theta_k + \frac{\delta\theta_k}{2}) \\ sin(\theta_k + \frac{\delta\theta_k}{2}) & \frac{\delta d_k}{2}cos(\theta_k + \frac{\delta\theta_k}{2}) \\ 0 & 1 \end{bmatrix} \tag{6}$$

The partial derivative of the control input u with respect to the parameters of a WFNN controller can be calculated by using the equations (7) and (8).

· Updating of WFNN parameters. The minimization is performed by the following iterative GD method.

$$\gamma_c(n+1) = \gamma_c(n) + \gamma_c(n) = \gamma_c(n) - \eta\frac{\partial C}{\partial \gamma} \tag{7}$$

where, η is the learning rate of a WFNN. From equations (5) and (6), $\frac{\partial u}{\partial \gamma_c}$ is the gradient of the controller output, u, with respect to parameters set γ_c.

$$\frac{\partial u_c}{\partial w_{jc}} = \frac{\partial \sum_{j=1}^{R} y_{jc}}{\partial w_{jc}}$$

$$= \frac{\prod_{n=1}^{3} \left(-\left(\frac{e_n - m_{k_n n}}{d_{k_n n}}\right)\right) exp\left(-\frac{1}{2}\left(\frac{e_n - m_{k_n n}}{d_{k_n n}}\right)^2\right)}{\sum_{r=1}^{R}\prod_{n=1}^{3} exp\left(-\frac{1}{2}\left(\frac{e_n - m_{k_n n}}{d_{k_n n}}\right)^2\right)}, \quad (8)$$

$$\frac{\partial u_c}{\partial a_{nc}} = e_n$$

where, subscript c and n denote control input and input of WFNN, respectively.

4. Simulation Results

In this section, we present simulation results to validate the control performance of proposed WFNN controller for the path tracking of mobile robots. In this simulation, we use parameters as shown in Table 1. Where, initial values of network weight are determined randomly. This simulation considers the tracking of a trajectory generated by the following displacements. :

$Linear\ velocity\ \delta d = 20cm/\text{sec.}\quad Angular\ velocity\ \delta\theta = 0°/\text{sec} \quad (0 < t \le 5)$
$Linear\ velocity\ \delta d = 30cm/\text{sec.}\quad Angular\ velocity\ \delta\theta = 59.3°/\text{sec} \quad (5 < t \le 10)$
$Linear\ velocity\ \delta d = 30cm/\text{sec.}\quad Angular\ velocity\ \delta\theta = -59.3°/\text{sec} \quad (10 < t \le 15)$
$Linear\ velocity\ \delta d = 20cm/\text{sec.}\quad Angular\ velocity\ \delta\theta = 0°/\text{sec} \quad (15 < t \le 20)$

Table 1. The parameters

Number of Membership Function (Number of Mother Wavelet)	11
Sampling Time	0.01
Learning Rate	0.2
Departure Posture vector	(5,5,0)

Fig. 4 shows the reference path and the controlled path using WFNN controller for a mobile robot. Also, Fig. 5 shows the control errors for path tracking of a mobile robot. Fig. 6 shows the control inputs from WFNN controller and the feed forward Jacobian of a mobile robot system.

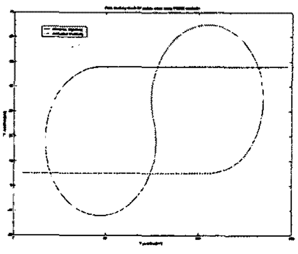


Fig. 4 Controlled path using WFNN controller

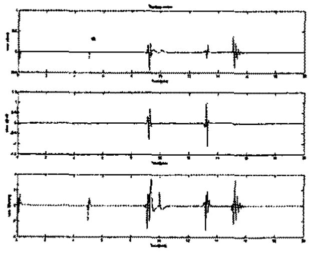


Fig. 5 Path tracking errors

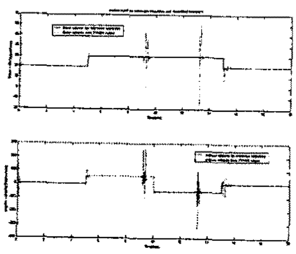


Fig. 6 Reference input and control input

From these figures and Table 2, we confirm that direct adaptive control system using our WFNN controller works well although the tracking error was occurred in case that a direction is changed.

Table 2. The simulation result

Mean-squared errors	x	0.0381 cm
	y	0.0432 cm
	θ	0.25 °

5. Conclusions

In this paper, we have proposed a FNN structure based on wavelet that merges the advantages of neural network, fuzzy model and wavelet. In addition, a WFNN controller based on direct adaptive control scheme has also been presented for the solution of the tracking problem for mobile robots. Through computer simulations, we have confirmed that direct adaptive control system using our WFNN controller works well although the tracking error was occurred shown in case that a direction is changed.

Acknowledgment

This work was supported by KOSEF R01-2001-000-00316

References

[1] C. C. Wit, H. Khennouf, C. Samson and O. J. Sordalen, "Nonlinear control design for mobile robots", Recent Trends in Mobile Robots, World Scientific, Singapore, pp. 121-156, 1993.

[2] J. M. Yang and J. H. Kim, "Sliding mode motion control of nonholonomic mobile robots", IEEE Control Systems, Vol. 19, No. 2, pp. 15-23, 1990.

[3] G. Dongbing and H. Huosheng, "Wavelet neural work based predictive control for mobile robots", Proc. of IEEE Int. Conf. on Systems, Man, and Cybernetics, Vol. 5, pp. 3544-3549, 2000.

[4] M. L. Corradini, G. Ippoliti, S. Longhi and S. Michelini, "Neural networks inverse model approach for the tracking problem of mobile robot", Proc. of RAAD 2000, pp. 17-22, 2000.

[5] M. L. Corradini, G. Ippoliti, and S. Longhi, "The tracking problem of mobile robots: experimental results using a neural network approach", Proc. of WAC pp. 33-37, 2000.

[6] S. Horikawa, T. Furuhashi and Y. Uchikawa, "On Identification of Structures in Premise of a Fuzzy Model Using a Fuzzy Neural Networks", Proc. of the 2nd IEEE Int. Conf. on Fuzzy Systems, pp. 661-666, 1993.

[7] T. Hasegawa, S. Horikawa, T. Furuhashi and Y. Uchikawa, "On Design of Adaptive Fuzzy Neural Networks and Description of Its Dynamical Behavior", Fuzzy Sets and Systems, Vol. 71, No. 1, pp. 3-23, 1995.