이동 로봇의 경로 추종을 위한 웨이블릿 신경 회로망 기반 예측 제어 시스템의 설계

*송 용 태, *박 진 배, **최 윤 호
*연세대학교 전기전자공학과, **경기대학교 전자공학부

Design of Wavelet Neural Network Based Predictive Control System for the Path Tracking of Mobile Robots

*Yong Tae Song, *Jin Bae Park, **Yoon Ho Choi
*Dept. of Electrical & Electronic Eng., Yonsei Univ., **School of Electronic Eng., Kyonggi Univ.

Abstract

본 논문에서는 이동 로봇의 경로 추종 제어를 위해 웨이블릿 신경 회로망에 기반한 예측 제어기의 설계 방법을 제안하고자 한다. 제안한 방법에 의해 설계된 제어기는 이동 로봇의 동특성을 예측하기 위한 웨이블릿 신경 회로망 기반 예측기와 예측 제어기로 구성된다. 제안한 방법에서 모델링 및 제어기로 적용되는 신경 회로망의 장점과 우수한 해석 능력을 가진 웨이블릿 변환의 장점을 결합한 웨이블릿 신경 회로망을 이용하여 이동 로봇의 동특성을 모델링하여 예측 제어기에서의 비용 함수 최소화에 적용한다. 경로 추종 제어의 목적인 이동 로봇의 실제 출력과 예측기의 출력 오차를 최소화하기 위해 웨이블릿 신경 회로망의 파라미터 동정 및 예측 제어기는 경사 하강법을 이용하여 학습한다. 마지막으로 컴퓨터 모의 실험을 통하여 제안한 예측 제어 시스템의 적용 가능성 및 효율성을 검증하고자 한다.

1. 서 론

이동 로봇에서의 경로 추종 문제는 최근의 논문들에서 많은 주목을 받아왔다. 특히 자율 이동성을 위한 정확한 경로 추종의 문제를 해결하기 위해 PID 제어, 슬라이딩 모드 제어, 예측 제어, 강인 제어등 다양한 제어 이론들을 기반으로 제어 기법들이 연구되어 왔다[1]-[2]. 그 중 신경 회로망은 임의적이며 연속적인 함수에 근사하는 고유한 능력 때문에 유용한 도구로서 사용되어 왔으나, 지능 제어 이론에 기반한 신경망 제어 방식은 국부 최소치(local minima)로의 수렴 가능성과 시스템의 동특성을 정확히 반영하지 못하는 단점을 갖고 있다. 따라서 시간 및 주파수 특성의 해석 능력이 우수한 웨이블릿 변환을 이용한 지능 제어 이론의 연구가 활발히 진행되고 있다[3]-[4]. 본 논문에서는 비선형성을 가지며 다중 제어 입력 및 다중 출력을 갖는 이동 로봇 시스템의 동특성을 정확히 반영한 지능 제어를 위해 일반형 예측 제어 기법에 기반한 웨이블릿 신경 회로망(Wavelet Neural Network) 제어 시스템의 설계 방법을 제안한다. 한편 설계된 웨이블릿 신경 회로망 제어기를 임의로 주어진 경로를 주행하는 이동 로봇 시스템에 적용하여 설계된 제어 시스템의 효율성 및 우수성을 검증하고자 한다.

2. 웨이블릿 신경 회로망

2.1 웨이블릿 신경 회로망의 구조

웨이블릿은 다음과 같이 모체(mother) 웨이블릿 함수로 구성된다.

$$\phi(z) = \phi\left(\frac{x-m}{d}\right) \tag{1}$$

여기서, m과 $d(>0)$는 각각 이동, 확장 인자를 나타내며, 모체 웨이블릿은 식(2)와 같은 가우시안(Gaussian) 함수의 1차 미분형이다.

$$\phi(z) = -z\exp\left(-\frac{1}{2}z^2\right) \tag{2}$$

그림 1은 N_i개의 입력과 1개의 출력을 갖는 WNN의 구조를 나타낸다.

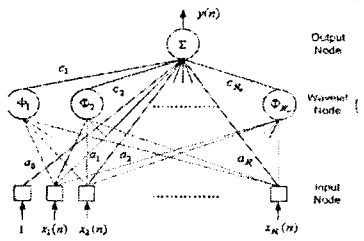

그림 1 웨이블릿 신경 회로망의 구조

그림 1에서 웨이블릿 노드의 각 웨이블릿은 식 (3)과 같이 각 입력에 대한 모체 웨이블릿의 곱으로 표현된다.

$$\Phi_j(\mathbf{x}) = \prod_{k=1}^{N_i}\phi(z_{jk}),\ \ with\ z_{jk} = \frac{x_k - m_{jk}}{d_{jk}} \tag{3}$$

여기서, $k=1,\cdots,N_i$, $j=1,\cdots,N_w$이며, N_i는 입력의 수, N_w는 웨이블릿 노드의 웨이블릿 수이다. 그림 1에서 WNN의 출력 값은 식 (4)같이 입력, 웨이블릿 및 각 파라미터들로 구성된다.

$$y = \Psi(\mathbf{x},\theta) = \sum_{j=1}^{N_w} c_j\Phi_j(\mathbf{x}) + a_0 + \sum_{k=1}^{N_i} a_k x_k \tag{4}$$

여기서, a_0와 a_k는 입력 노드와 출력 노드 사이의 가중치(weight), c_j는 웨이블릿 노드와 출력 노드 사이의 가중치이고 θ는 학습을 통하여 갱신되는 WNN의 파라미터 집합이다.

$$\theta = [a_0, a_k, c_j, m_{jk}, d_{jk}] \tag{5}$$

2.2 웨이블릿 신경 회로망의 학습 방법

WNN의 학습은 식 (6)과 같은 비용 함수를 최소화하는 파라미터 집합 θ를 구하는 것이다.

$$J(\theta(n)) = \frac{1}{2}(y_r(n) - y(n))^2 = \frac{1}{2}e^2(n) \tag{6}$$

여기서, $y(n)$은 n 번째 WNN의 출력이며, $y_r(n)$은 $y(n)$의 기준 출력 값이다. 파라미터 집합 θ는 식 (7)과 같은 경사 하강법(gradient descent method)을 이용하여 구해진다.

$$\theta(n+1) = \theta(n) - \Delta\theta(n) = \theta(n) - \eta\frac{\partial J(\theta(n))}{\partial\theta(n)} \tag{7}$$

여기서, η는 WNN의 학습률(learning rate)이고, $\frac{\partial J(\theta(n))}{\partial \theta(n)}$은 $\theta(n)$에 대한 비용 함수의 기울기로서 식 (8)에 의해 계산된다.

$$\frac{\partial J(\theta(n))}{\partial \theta(n)} = -e(n)\frac{\partial y(n)}{\partial \theta(n)} \tag{8}$$

식 (8)에서 $\theta(n)$에 대한 $y(n)$의 기울기 $\frac{\partial y(n)}{\partial \theta(n)}$는 출력을 각각의 파라미터에 대해 미분하여 구해진다.

3. 웨이블릿 신경 회로망을 이용한 예측 제어기 설계

3.1 이동 로봇 모델

본 논문에서 사용한 이동 로봇은 두 개의 구동륜 및 네 개의 캐스터로 구성되었으며, 이동 로봇의 위치는 두 개의 구동륜 사이의 중심 좌표에 의해 표현되어지는 3자유도로 고려되어 진다. 또한 이동 로봇의 방위각은 그림 2와 같이 표현되며, 이동 로봇의 동적 방정식은 식 (14)과 같다.

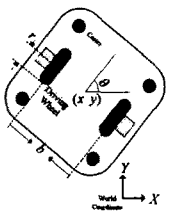

그림 2 이동 로봇 모델 및 절대좌표

$$\begin{pmatrix} X_{k+1} \\ Y_{k+1} \\ \theta_{k+1} \end{pmatrix} = \begin{pmatrix} X_k \\ Y_k \\ \theta_k \end{pmatrix} + \begin{pmatrix} \delta d_k \cos(\theta_k + \frac{\delta\theta_k}{2}) \\ \delta d_k \sin(\theta_k + \frac{\delta\theta_k}{2}) \\ \delta\theta_k \end{pmatrix}, \quad \delta d = \frac{d_r - d_l}{2} \quad \delta\theta = \frac{d_r - d_l}{2} \tag{14}$$

여기서 δd 및 $\delta\theta$은 직진 속도 및 각 속도를 나타내며 이동 로봇의 모델에서 제어 입력 $u_k = [\delta d_k, \delta\theta_k]^T$ 로 정의된다. 또한 d_r, d_l 및 b는 각각 좌·우 구동륜의 변위 및 구동륜 사이의 거리를 나타낸다.

3.2 웨이블릿 신경 회로망을 이용한 예측 제어 시스템

본 논문에서 제안한 직접 적응제어 기법에 기반한 웨이블릿 신경 회로망 예측 제어 시스템의 구조는 그림 3과 같다.

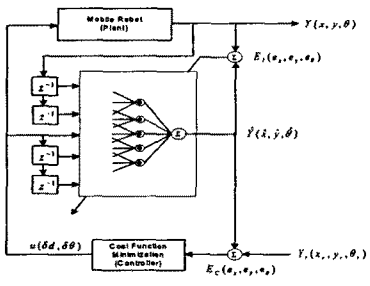

그림 3 웨이블릿 신경 회로망을 이용한 일반형 예측 제어 시스템의 구조

그림에서 기준 출력과 웨이블릿 신경 회로망 출력의 차로 정의되는 $E_C(e_x, e_y, e_\theta)$은 제어기의 파라미터를 조정하여 시스템 제어 오차를 최소화 하며, 이와 같이 제어 오차를 최소화 함으로써 시스템 출력 $Y(x, y, \theta)$가 기준 신호 $Y_r(x_r, y_r, \theta_r)$을 추종하도록 한다. 플랜트가 다중 입출력 시스템일 경우 플랜트의 모델식 (14)에 따라 다음과 같이 웨이블릿 신경망을 구성한다.

$$\begin{aligned}
\hat{x}(n) &= f(x(n-1), \theta(n-1), \cdots, x(n-N_s), \theta(n-N_s), \\
&\quad u_{\delta d}(n-2), u_{\delta\theta}(n-2), \cdots, u_{\delta d}(n-N_c), u_{\delta\theta}(n-N_c)) \\
\hat{y}(n) &= f(y(n-1), \theta(n-1), \cdots, y(n-N_s), \theta(n-N_s), \\
&\quad u_{\delta d}(n-2), u_{\delta\theta}(n-2), \cdots, u_{\delta d}(n-N_c), u_{\delta\theta}(n-N_c)) \\
\hat{\theta}(n) &= f(\theta(n-1), \cdots, \theta(n-N_s), u_{\delta\theta}(n-2), \cdots u_{\delta\theta}(n-N_c))
\end{aligned} \tag{15}$$

여기서 $\hat{x}(n)$, $\hat{y}(n)$, $\hat{\theta}(n)$은 n번째 웨이블릿 신경 회로망의 출력이고 $f(\cdot)$은 실제 시스템의 비선형 함수이며 N_s와 N_c는 각각 웨이블릿 신경회로망의 상태 입력의 수와 제어 입력의 수를 나타낸다. 일반형 예측 제어기의 설계를 위한 비용 함수는 식 (16)과 같이 정의한다.

$$J = \frac{1}{2}(E_C^2(n+1) + \lambda\Delta u^2(n)) \tag{16}$$

식 (16)의 $E_C^2(n+1)$은 제어 오차로서 다음과 같이 정의한다.

$$E_C^2(n+1) = \hat{Y}(n+1) - Y_r(n+1) \tag{17}$$

여기서 $Y_r(n+1)$은 기준 신호이다.
웨이블릿 신경 회로망을 통과한 시스템 출력은 식 (18)과 같이 얻을 수 있다.

$$\hat{Y} = \sum_{j=1}^{N_w} c_j \Phi_j(\mathbf{x}) + a_0 + \sum_{k=1}^{N_s} a_k x_k \tag{18}$$

식(16)의 비용 함수 J를 최소화 하기 위해서 제어 신호 $u(n)$은 경사하강법에 의해 반복적으로 계산되며, 식 (19)에 의해 갱신된다.

$$\begin{aligned}
u_{\delta d}(n+1) &= u_{\delta d}(n) - \eta\frac{\partial J}{\partial u_{\delta d}(n)} \\
u_{\delta\theta}(n+1) &= u_{\delta\theta}(n) - \eta\frac{\partial J}{\partial u_{\delta\theta}(n)}
\end{aligned} \tag{19}$$

식(19)의 $\frac{\partial J}{\partial u_{\delta d}(n)}$, $\frac{\partial J}{\partial u_{\delta\theta}(n)}$은 제어 신호 $u_{\delta d}(n), u_{\delta\theta}(n)$에 대한 비용 함수의 편미분으로 식(20)에 의해 구해진다.

$$\begin{aligned}
\frac{\partial J}{\partial u_{\delta d}(n)} &= [(\hat{x}(n+1) - x(n+1))\frac{\partial \hat{Y}_x(n+1)}{\partial u_{\delta d}(n)} \\
&\quad + (\hat{y}(n+1) - y(n+1))\frac{\partial \hat{Y}_y(n+1)}{\partial u_{\delta d}(n)}] + \lambda\Delta u_{\delta d}(n) \\
\frac{\partial J}{\partial u_{\delta\theta}(n)} &= [(\hat{\theta}(n+1) - \theta(n+1))\frac{\partial \hat{Y}_\theta(n+1)}{\partial u_{\delta\theta}(n)}] + \lambda\Delta u_{\delta\theta}(n)
\end{aligned} \tag{20}$$

또한 식 (20)의 $\frac{\partial \hat{Y}(n+1)}{\partial u(n)}$은 제어 신호 $u(n)$에 대한 웨이블릿 신경 회로망의 기울기로서 식 (21)로 나타낼 수 있다.

$$\begin{aligned}
\frac{\partial \hat{Y}(n+1)}{\partial u(n)} &= \sum_{j=1}^{N_w} c_j \frac{\partial \Phi_j(\mathbf{x})}{\partial u(n)} + a_k \mid_{k=N_s+1} \\
&= \sum_{j=1}^{N_w} \frac{c_j}{d_k} \frac{\partial \Phi_j(\mathbf{x})}{\partial \mathbf{x}} \frac{\partial \mathbf{x}}{\partial u(n)} + a_k \mid_{k=N_s+1}
\end{aligned} \tag{21}$$

따라서 제어 신호 $u(n)$은 다음과 같이 나타낼 수 있다.

$$u_{\delta d}(n+1) = u_{\delta d}(n) - \eta \frac{\partial J}{\partial u_{\delta d}(n)}$$
$$= u_{\delta d}(n) - \eta \left[\sum_{j=1}^{N_w} \frac{c_j}{d_{jk}} \frac{\partial \Phi_j(\mathbf{x})}{\partial \mathbf{x}} \frac{\partial \mathbf{x}}{\partial u_{\delta d}(n)} + a_k|_{k=N_i+1} + \lambda \Delta u_{\delta d}(n) \right]$$

$$u_{\delta\theta}(n+1) = u_{\delta\theta}(n) - \eta \frac{\partial J}{\partial u_{\delta\theta}(n)}$$
$$= u_{\delta\theta}(n) - \eta \left[\sum_{j=1}^{N_w} \frac{c_j}{d_{jk}} \frac{\partial \Phi_j(\mathbf{x})}{\partial \mathbf{x}} \frac{\partial \mathbf{x}}{\partial u_{\delta\theta}(n)} + a_k|_{k=N_i+1} + \lambda \Delta u_{\delta\theta}(n) \right] \quad (22)$$

플랜트 모델 식 (14)에 따라 제어 신호의 관계를 수식적으로 연관성이 있는 입출력끼리 구분하여 구성하고, 제어 신호를 생성시키는 제어기 또한 모델식에 따라 인자들을 구분한다. 따라서 식 (22)의 $\frac{\partial \mathbf{x}}{\partial u(n)}$도 각각의 웨이블릿 신경망 입력 벡터를 $\mathbf{x}_x, \mathbf{x}_y, \mathbf{x}_\theta$ 라고 한다면 다음과 같이 쓸 수 있다.

$$\frac{\partial \mathbf{x}_x}{\partial u_{\delta d}} = [0\,0\,1\,0]^T, \frac{\partial \mathbf{x}_y}{\partial u_{\delta d}} = [0\,0\,1\,0]^T, \frac{\partial \mathbf{x}_\theta}{\partial u_{\delta d}} = [0\,0]^T$$
$$\frac{\partial \mathbf{x}_x}{\partial u_{\delta\theta}} = [0\,0\,0\,0]^T, \frac{\partial \mathbf{x}_y}{\partial u_{\delta\theta}} = [0\,0\,0\,0]^T, \frac{\partial \mathbf{x}_\theta}{\partial u_{\delta\theta}} = [0\,1]^T$$

4. 모의 실험

4.1 모의 실험 환경

본 논문에서는 기존의 단일 입출력 기반 웨이블릿 예측 제어 기법을 다중 입력, 다중 출력 시스템인 이동 로봇 모델에 적용하여 경로 추종이 성공적으로 수행됨을 확인함으로써, 웨이블릿 신경망 기반 예측 제어 시스템이 일반적인 시스템에 성공적으로 적용될 수 있음을 보이고자 한다. 모의 실험 환경은 로봇 궤적 설정과 예측기 파라미터를 설정하게 되며, 초기 연결 강도는 랜덤하게 발생한다. 이동 로봇의 궤적은 표 1과 같이 설정하였다.

표 1 이동 로봇 궤적 설정

Time	Linear velocity(δd)	Angular velocity($\delta\theta$)
$0 < t \le 5$	$0.1m/sec$	$0^\circ/sec$
$5 < t \le 10$	$0.2m/sec$	$5.93/sec$
$10 < t \le 15$	$0.2m/sec$	$-5.93/sec$
$15 < t \le 20$	$0.1m/sec$	$-0.593/sec$

또한 웨이블릿 신경망은 표 2과 같이 구성하였다.

표 2 WNN 예측제어 시스템의 모의 실험 초기 조건

Number of wavelet function (predictor)	5
Number of past state (controller)	1
Number of past input (controller)	1
Sampling time (controller)	0.01

4.2 모의 실험 결과

표 3에서와 같이 시스템 동정 및 기준 신호에 대해 x, y, θ 모두 양호한 수준의 평균 제곱 오차를 보여준다. 또한 그림 4에서처럼 이동 로봇의 비선형적인 궤적을 상당히 정확하게 추종함을 알 수 있다. 그림 5는 이동로봇의 제어 입력 신호를 나타내는데, 기준 궤적을 생성시키는 제어 신호에 근사한 값이 생성됨을 보여준다.

표 3 평균 제곱 오차

MSE(Model-Plant)		MSE (Reference-Plant)	
x	0.0089cm	x	0.3916cm
y	0.1178cm	y	0.3090cm
θ	0.3094 °	θ	0.0172 °

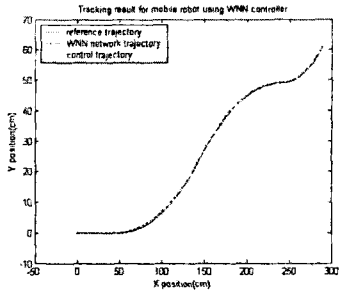

그림 4 웨이블릿 신경 회로망 기반 예측 제어 시스템의 경로 추종 결과

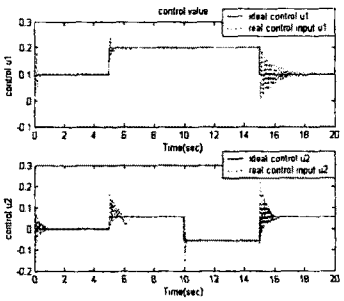

그림 5 제어 입력

5. 결 론

본 논문에서는 다중 입력 및 다중 출력을 가지는 비선형 시스템의 하나인 이동 로봇의 경로 추종 제어를 위하여 웨이블릿 신경 회로망을 이용한 예측 제어 시스템의 설계 방법을 제안하였다. 웨이블릿 신경 회로망 모델과 제어기의 학습 방법은 각각 경사 하강법을 사용하였다. 모의 실험 결과에서 보여지듯이 본 논문에서 제안된 방법의 적용 가능성과 효율성을 검증하기 위하여 임의로 주어진 경로를 추종하게 한 결과, 수렴 속도와 평균 제곱 오차(MSE) 면에서 양호한 제어 성능을 보여 단일 입출력 뿐만 아니라 다중 입출력에도 웨이블릿 신경망 기반 예측 제어 시스템이 적용될 수 있음을 확인하였다.

감사의글: 본 논문은 한국과학재단의 목적기초연구사업(R01-2001-000-00316)에 의해 지원되었습니다.

[참 고 문 헌]

[1] X. Yang, K. He, M. Guo, and B. Zhang, "An Intelligent Predictive Control Approach to Path Tracking Problem of Autonomous Robot", *Proc. of IEEE Conf. on System, Man, and Cybernetics*, Vol. 4, pp. 350-355, 1998.

[2] Z. P. Jiang and H. Nijmeijer, "Tracking Control of Mobile Robots: A Case Study in Backstepping", *Automatica*, Vol. 33, pp. 1393-1399, 1997.

[3] Y. Oussar, I. Rivals, L. Personnaz and G. Dreyfus, "Training Wavelet Networks for Nonlinear Dynamic Input-Output Modeling", *Neurocomputing*, Vol. 20, pp. 173-188, 1998.

[4] Q. Zhang and A. Benveniste, "Wavelet Networks", *IEEE Trans. on Neural Network*, Vol. 3, pp. 889-898, 1992.