

뉴럴-퍼지제어기법에 의한 두 구동휠을 갖는 이동형 로봇의 자세 및 속도 제어

The Azimuth and Velocity Control of a Mobile Robot with Two Drive Wheels by Neural-Fuzzy Control Method

조 용 길 · 배 종 일

Y. G. Cho and J. I. Bae

Key Words : Azimuth(방향각), Back Propagation Algorithm(역전파 알고리즘), Neural-Fuzzy Control(뉴럴-퍼지제어), Membership Function(멤버쉽함수), Supervised Learning(지도학습), Reinforcement Learning(보강학습)

Abstract : This paper presents a new approach to the design of speed and azimuth control of a mobile robot with two drive wheels. The proposed control scheme uses a Gaussian function as a unit function in the neural-fuzzy network and back propagation algorithm to train the neural-fuzzy network controller in the framework of the specialized learning architecture. It is proposed to a learned controller with two neural-fuzzy networks based on an independent reasoning and a connection net with fixed weights to simplify the neural-fuzzy network. The performance of the proposed controller can be seen by the computer simulation for trajectory tracking of the speed and azimuth of a mobile robot driven by two independent wheels.

1. 서 론

기존의 이동형 로봇의 제어기는 제어하고자 하는 물체의 운동방정식을 알고 있을 때만 가능하고, 운동방정식은 비선형성을 포함하고 있으므로 복잡하고, 정확히 구하지 못하는 경우가 많다¹⁻³⁾.

또한, 널리 사용되는 고전적 제어의 경우 이득값이 고정되어 있고, 이러한 제어방법은 외부 환경의 변화에 대응할 수 없으므로 제어성능에 한계점을 드러내고 있다. 최근, 뉴럴-퍼지(Neural-fuzzy)이론을 제어시스템에 적용함으로써 고전적 제어이론의 한계성 극복에 커다란 가능성을 보이고 있다.

뉴럴-퍼지제어방법은 불확실성 시스템의 특성을 지니는 제어계에 적용되고, 이동형 로봇의 제어는 정확한 수학적인 동적 모델링이 요구되지 않으므로 선형시스템은 물론 비선형시스템에도 쉽게 적용할 수 있는 장점을 지니고 있다. 이러한 장점을 이용하여 파라미터의 동특성이 복잡한 비선형

시스템을 제어하기에 적절한 제어기법으로서 많은 연구가 진행되고 있다. 퍼지제어이론을 이용한 제어기의 설계시에는 입·출력 데이터 및 멤버쉽함수의 수와 범위에 관하여 제어기의 파라미터들을 맞추어야 하고, 신경회로망 이론을 이용한 제어는 기율기법 또는 역전파 알고리즘을 이용하여 모델링과 추정을 할 수 있는 특성이 있으며, 집합적 처리능력은 복잡한 지각입력들에 대한 빠른 응답을 제공해 주는 특징이 있다. 신경망 제어기법의 특징은 입·출력 정보가 불확실하더라도 신경망 학습기능으로 인하여 좋은 역동력학적 효과가 있고, 비선형성, 학습, 병렬처리⁴⁾ 등의 장점이 있다. 뉴럴-퍼지제어기의 기본원리는 신경망위 세포의 학습함수를 이용한 퍼지제어규칙과 멤버쉽함수를 자동적으로 조정하는 제어방법이라 할 수 있으며, 전문가에 의하여 구동되는 시스템으로 부터 처리된 패턴 데이터로서 제어규칙을 확인할 수 있고, 파라미터들을 정확히 맞춘다. 이에 대표적으로 호리카와에 의한 방법^{5,6)}이 일반화된 학습 구조가 신경망을 다루는데 사용되고, 제어데이터가 없는 경우에는 적용할 수 없다. 다시 말하면, 이 제어구조의 접근

접수일 : 1998년 9월 15일
조용길 : 동아대학교 대학원
배종일 : 부경대학교 전기공학과

방법은 유사 사다리꼴류 함수가 다른 기호를 가진 두개의 시그모이드 단위함수(Sigmoidal unit function)를 합한 것에 의해서 구성되기 때문에 선행단계에서의 멤버쉽함수를 산출하기 위한 중간층(Intermediate layer)이 요구되는 방법이다. 이와 같은 방법을 퍼지 시그모이드 신경망이라 하며, 연결 중간층에서 단일체(Unity)의 수는 퍼지 표시들의 수가 증가함으로써 성장한다. 본 연구에서는 이동형 로봇의 속도 및 방향제어를 위하여 역전파 알고리즘에 의한 다층신경망 구조와 가우스 멤버쉽함수의 혼합 퍼지추론법을 적용한 새로운 구조의 뉴럴-퍼지(Neural-fuzzy, NF)제어기법을 제안하고, 제안된 뉴럴-퍼지제어기법에 의해 설계된 이동형 로봇 제어기의 성능을 컴퓨터 시뮬레이션을 통하여 예증한다.

2. 뉴럴-퍼지 제어기

2.1 추론방법 및 멤버쉽함수

본 연구의 퍼지추론은 수계노(Sugeno)의 퍼지추론⁷⁾의 단순화된 혼합 추론법이다. 사실, 뉴럴-퍼지 제어기법은 수계노의 퍼지추론의 끝단 함수가 상수가 되거나 또는 최소-최대 도심법의 끝단에서 퍼지집합의 범위가 미소값, 즉 싱글톤(Singleton)이 될 때와 일치한다. 따라서, 임의의 i 번째 제어규칙은 다음과 같이 쓸 수 있다.

$$H_i : \text{If } x_1 = M_{i1} \text{ and ... and } x_n = M_{in} \\ \text{then } \Gamma_1 = N_{i1} \text{ and ... and } \Gamma_n = N_{in} \quad (1)$$

여기서, H_i 는 i 번째 제어규칙이고, M_{ij} 는 i 번째 제어규칙에서 j 번째 입력변수와 결합되는 선행단계에서의 퍼지집합이다. N_{ij} 는 i 번째 제어규칙의 끝단에서의 j 번째 변수와 결합되는 상수이다.

신뢰도 $\sigma M_{i1}(x_1), \dots, \sigma M_{in}(x_n)$ 을 적용하면, 선행단계에서의 신뢰도 Q_i 는 다음과 같이 정의된다.

$$Q_i = \sigma M_{i1}(x_1) \cdot \sigma M_{i2}(x_2) \cdot \dots \cdot \sigma M_{in}(x_n) \quad (2)$$

j 번째 출력 결과는 가중치 Q_i 에 관하여 N_{ij} 의 가중 평균치로서 다음 식과 같다.

$$\Gamma_j^* = \frac{\sum_{i=1}^r Q_i N_{ij}}{\sum_{i=1}^r Q_i}, \quad j = 1, \dots, n \quad (3)$$

멤버쉽함수는 다층의 신경망 내의 단위함수에 대하여 $[0,1]$ 구간을 갖는 함수로 정의된다.

$$F(\mu x) = (e^{\mu x} + 1)^{-1} \quad (4)$$

여기서, μ 는 영보다 작은 상수이다.

Fig. 1은 신경망을 이용한 구조이며, 기호 { }를 가진 변수는 신경망을 통한 신호를 나타낸다. 원형 기호는 단일체이고, Ω_i 는 입력변수 x_i 에 대한 계수이고, Ω_c 와 Ω_d 는 각각 중심값과 시그모이드 함수의 구배를 나타내는 연결 가중치이다. 기호 -1을 갖는 단일체는 -1의 결과를 산출하고, Σ 기호를 가지는 단일체는 입력들의 합을 산출한다. 그리고 기호 F를 갖는 단일체에서의 입·출력 관계는 식(4)에 의해서 정의되고, 기호를 가지지 않는 단일체는 입력을 출력으로 간단히 산출한다. Fig. 1에서 신경망을 적용하면, 영에서 중심을 갖는 함수는 식(5)와 음의 값을 부과함으로써 구성할 수 있다.

$$F(x_i) = [e^{a(bx_i + c)}]^{-1} \quad (5)$$

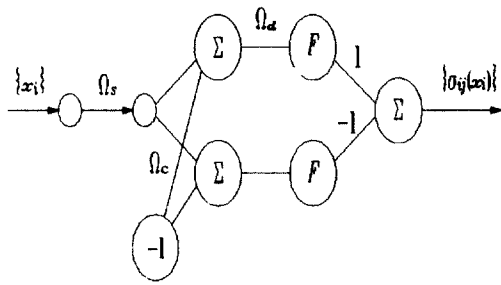
여기서, a, b 및 c 는 임의의 상수를 나타낸다.

퍼지 시그모이드 신경망은 유사 사다리꼴류 함수가 다른 기호를 가진 두개의 시그모이드 단위함수의 합으로 구성되기 때문에, 선행단계에서 멤버쉽함수를 산출하기 위한 중간층들을 생각할 필요가 있다. 이러한 사실은, 연결 중간층에서 단일체의 수는 퍼지 라벨의 수가 증가함으로써 성장하고, 뉴럴-퍼지법은 단위 함수로서 정의되는 가우스함수로 정의한다.

$$F(x) = e^{\ln(1/2)x^2} \quad (6)$$

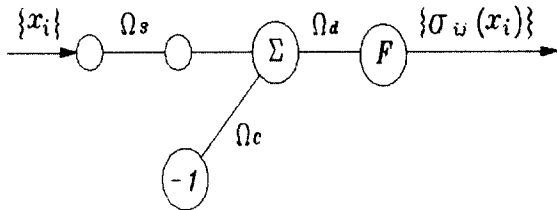
신경망을 이용한 멤버쉽함수의 구성은 Fig. 2이고, 연결 가중치 Ω_c 는 가우스함수에 대한 중심값을 나타내고, Ω_d 는 표준화된 후원집합에서 가우스함수 0.5를 갖는 중심 Ω_c 로부터 편차의 역으로 나타내며, 신경망과 식(6)을 이용하면 다음 식과 같다.

$$F(x_i) = e^{\ln(1/2) [\Omega_d(\Omega_c - x_i)]^2} \quad (7)$$



(I) (II) (III) (IV) (V)

Fig. 1 Neural network for generating a membership function



(I) (II) (III) (IV)

Fig. 2 Neural network for gaussian membership function

2.2 뉴럴-퍼지 제어기의 구조 및 원리

Fig. 3은 두개의 입력 (x_1, x_2)와 하나의 출력 (Γ_1^*)과 선행단에서 세개의 라벨들이 있는 경우에 대하여 뉴럴-퍼지의 구성도이다. 이때, 제어규칙 수는 $r=32$ 이고, 그림 (VI)층에 대하여 Σ 와 P의 기호를 갖는 단일체는 선형 덧셈 입력을 갖는다.

$$P(x) = x^{-1} \tag{8}$$

Fig. 3에서 (I) ~ (V)층은 퍼지제어규칙의 선행단에 해당하고 (VII)과 (VIII)층은 끝단에 해당한다. (I)층에 적용된 입력 x_i 는 연결가중치 Ω_s 와 비례하며 (III)층의 바이어스인 연결가중치 Ω_c 는 (IV)

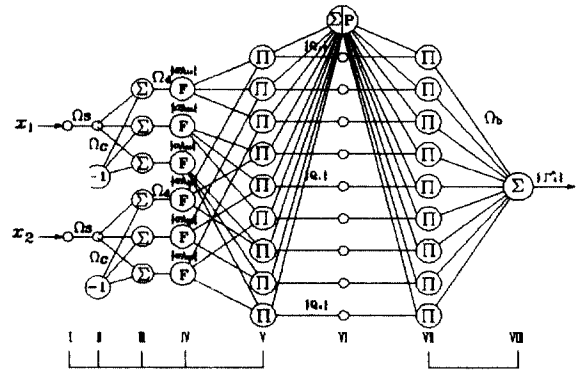


Fig. 3 Basic structure of neural-fuzzy network

층에서 가우스 함수에 대한 입력인 Ω_d 에 의해 곱하여지고, (V)층에서는 모든 제어규칙들에 대한 선행단에서 신뢰도 Q_i 를 얻고, (VI)층의 첫번째 단일체 입력들의 합과 역 계산을 수행한다. 마지막으로, (VII)층과 (VIII)층에서 Q_i 에 관한 Ω_b 의 평균치로서 결과값을 얻는다.

역전파 알고리즘을 적용함으로써 뉴럴-퍼지 제어기의 연결 가중치들을 학습시킬 수 있고, 제어규칙들을 검증할 수 있으며 선행단의 멤버쉽함수를 맞출 수 있다. 그러나, 일반화된 학습구조 및 퍼드백 오차 학습구조에 기초를 둔 경우는 결과층에서 η 값을 변화시킨다. 식(10)은 k 번째층에 대한 입력이 함수 $F(\cdot)$ 를 통한 출력이고, $(k+1)$ 번째층에 대한 입력이 덧셈작업에 의해 계산될 때의 경우로 나타낸다. 식(11)은 k 번째층에 대한 입력이 함수 $F(\cdot)$ 를 통한 출력이고, $(k+1)$ 번째층에 대한 입력이 대수적에 의해서 표시된다고 하면, 출력층 q 에서의 j 번째 단일체내의 η_j^q 와 어떤 중간층 k 에서 j 번째 단일체내의 η_j^k 는 각각 식(9), 식(10) 및 식(11)로 주어진다. η 는 원하는 출력 (x_{di})과 실제 출력 (x_i)과의 오차 활성화치 함수의 미분량 및 구동입력의 출력에 대한 Jacobian식으로 나타낸다.

$$\eta_j^q = F'(I_j^q) \sum_{i=1}^m (x_{di} - x_i) \frac{\partial x_i}{\partial I_j^q} \tag{9}$$

$$\eta_j^k = F'(I_j^k) \sum \eta_i^{k+1} \Omega_{ji}^{k+1} \tag{10}$$

$$\eta_j^k = F(I_j^k) \sum \eta_i^{k+1} \Omega_{ij}^{k,k+1} (\prod_{i \neq j} \Omega_{ij}^{k,k+1} z_i^k) \quad (11)$$

여기서, 식(9)는 출력층 q에 대한 η 값을 나타내고, 식(9)와 식(10)은 중간층에 대한 η 의 값을 계산하고 있으므로, η 값에 대하여 식(10)의 적용은 (IV)와(VI)층에서 계산된다.

한편, (IV)와(VI)층의 첫번째 단일체에서의 F' 는 다음 식과 같다.

$$F'(I_j^k) = \ln(1/2)(I_j^k z_j^k)^2 \quad (12)$$

$$F'(I_j^k) = -(z_j^k)^2 \quad (13)$$

식(12)는 (IV)층에 대한 값이고, 식(13)은 (VI)층의 첫번째 단일체에 대한 값이다.

그리고, 다른 선형 단일체에 대하여 $F'(I_j^k) = 1$ 로 계산한다.

$$\text{또한, } \frac{\partial x_i(kT)}{\partial \Gamma_j(kT)} \approx \frac{\Delta x_i(kT)}{\Delta \Gamma_j(kT)} \quad \text{로서 식(10)}$$

의 Jacobian $\partial x_i / \partial \Gamma_j$ 의 값을 구한다.

여기서, $\Delta \Gamma_j(\cdot)$ 와 $\Delta x_i(\cdot)$ 는 입·출력데이터이고, $\Delta = 1 - d^{-1}$ 로 부터 산출된다.

그리고 d^{-1} 는 한 스텝지연 동작이고, k 는 이산 시간이고 T는 샘플링 범위이다.

만약 시스템이 입력의 어떤 시간 지연을 갖지 않는 최초의 이산시간 지연 시스템이라면 위의 방정식 대신 $\partial x_i(kT) / \partial \Gamma_j[(k-1)T]$ 값을 의미한다.

이러한 결과의 연결 가중치들은 식(14)와 식(15)로 나타낼 수 있다.

$$\Omega_{ij}^{k-1,k}(t+1) = \Omega_{ij}^{k-1,k}(t) + \xi \eta_j^k z_i^{k-1} + b \Delta \Omega_{ij}^{k-1,k}(t) \quad (14)$$

$$\Omega_{ij}^{k-1,k}(t+1) = \Omega_{ij}^{k-1,k}(t) + \xi \eta_j^k z_i^{k-1} + (\prod_{i \neq j} \Omega_{ij}^{k-1,k} z_i^{k-1}) + b \Delta \Omega_{ij}^{k-1,k}(t) \quad (15)$$

여기서, t는 k번째 갱신시간을 나타내고, ξ 는 학습비를 의미하는 작은 양의 상수이다.

$\Delta \Omega_{ij}^{k-1,k}(t)$ 는 t번째 구간에서 연결 가중치의 증분이고, b는 안정요소를 의미하는 작은 양의 상수이며, 식(14)를 이용하여 연결 가중치 $\Omega_s, \Omega_c, \Omega_d, \Omega_b$ 로 나타낸다.

식(15)와 관련된 연결 가중치들이 모두 단일체로서 고정되어 있으므로, 어떤 연결 가중치들의 갱신방정식을 사용하지 않는다. 또한 입력층에서 두 입력신호 x_1 및 x_2 가 서로 다른 경우 척도요소(Scaling factor) Ω_s 도 매우 크게 달라지게 된다.

그러므로, 보통 척도요소를 두 종류로 분리하여 설정하여야 되므로 그 문제점을 해결하기 위한 연결 가중치 Ω_s 의 변형된 식은 다음과 같다.

$$\Omega_{ij}^{k-1,k}(t+1) = \Omega_{ij}^{k-1,k}(t) + \xi \eta_j^k \Omega_{ij}^{k-1,k}(t)^2 z_i^{k-1} + b \Delta \Omega_{ij}^{k-1,k}(t) \quad (16)$$

2.3 뉴럴 퍼지 제어기의 학습구조

본 논문에서는 구동토크와 시스템의 출력 정보 사이의 관계식을 얻기 위하여 제어기의 영역에서 폭넓게 사용되는 Computer aided torque controller를 서보제어기의 모델기준으로 활용하였으며, 설계기법으로 부터 얻어진 제어원리와 기본구조가 두개의 뉴럴-퍼지로 구성되는 학습제어기에 적용한다. 이동형 로봇의 동적 모델링식은 다음과 같이 표현할 수 있다.⁸⁾

$$\begin{aligned} 2 \Gamma_r &= \frac{\ddot{\theta}^*}{f_k R L (M_v R^2 + 2 M_w L^2)^{-1}} \\ &- \frac{-2 f_c L^2 (M_v R^2 + 2 M_w L^2)^{-1} \dot{\theta}}{f_k R L (M_v R^2 + 2 M_w L^2)^{-1}} \\ &+ \frac{\dot{V}^*}{f_k R (m R^2 + 2 M_w)^{-1}} \\ &- \frac{-2 f_c V (m R^2 + 2 M_w)^{-1}}{f_k R (m R^2 + 2 M_w)^{-1}} \\ 2 \Gamma_l &= - \frac{\ddot{\theta}^*}{f_k R L (M_v R^2 + 2 M_w L^2)^{-1}} \\ &- \frac{-2 f_c L^2 (M_v R^2 + 2 M_w L^2)^{-1} \dot{\theta}}{f_k R L (M_v R^2 + 2 M_w L^2)^{-1}} \end{aligned} \quad (17)$$

$$\begin{aligned}
 & + \frac{\dot{V}^*}{f_k R (mR^2 + 2M_w)^{-1}} \\
 & - \frac{-2f_c V (mR^2 + 2M_w)^{-1}}{f_k R (mR^2 + 2M_w)^{-1}}
 \end{aligned} \tag{18}$$

여기서, $\ddot{\theta}^*$ 는 $\ddot{\theta}_d$ 에 PD서보를 더하는 것으로 산출되고, \dot{V}^* 는 \dot{V}_d 에 P서보를 더하는 것으로 얻는다.

$$\ddot{\theta}^* = \ddot{\theta}_d + K_{\theta V} \dot{\delta}_\theta + K_{\theta P} \delta_\theta \tag{19}$$

$$\dot{V}^* = \dot{V}_d + K_{VP} \delta_V \tag{20}$$

여기서, $K_{\theta V}$ 와 $K_{\theta P}$ 는 각각 속도오차에 대한 미분게인 및 위치오차에 대한 비례게인을 나타내고, 출력 편차인 δ_θ 와 δ_V 는 식(21)과 식(22)로 정의된다.

$$\delta_\theta = \theta_d - \theta \tag{21}$$

$$\delta_V = V_d - V \tag{22}$$

여기서, θ_d 와 V_d 는 각각 방향각과 기준속도이다. 이동형 로봇에 대한 속도와 편차는 양 구동바퀴를 구동하는데 사용되지만, 각 구동바퀴에서 똑같은 정보량을 가지지는 않는다. 반면 방향각의 비와 각 구동바퀴의 편차와 방향각의 편차는 두 구동바퀴를 구동하는데 사용되지만, 각 구동바퀴에서 반대기호의 정보량을 가지지는 않는다.

제안된 제어기에서 모델기준 서보제어의 구조를 분석하면, $\Gamma_r = \Gamma_V + \Gamma_\theta$, $\Gamma_l = \Gamma_V - \Gamma_\theta$ 의 관계식이 주어짐을 알 수 있다.

Γ_V 는 이동형 로봇의 속도를 제어하기 위하여 요구되는 토크이고, Γ_θ 는 방향각을 제어하기 위한 토크이다.

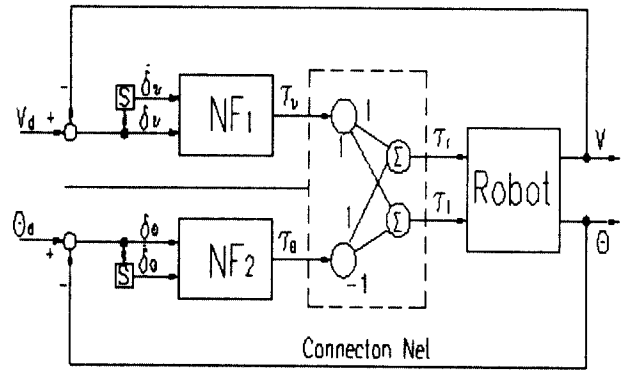


Fig. 4 The global structure of neural-fuzzy controller

3. 시뮬레이션

3.1 이동형 로봇의 동적 모델링

두 개의 독립 구동바퀴를 갖는 이동형 로봇가 평면상에서 정확히 이동하고, 절대 좌표계 O-XY는 평면상에 고정되었다고 가정한 상태에서 운동 방정식은 다음 식과 같다.

$$\begin{aligned}
 M_V \ddot{\theta} + (-T_r + T_l)L &= 0 \\
 m \dot{V} - (T_l + T_r) &= 0
 \end{aligned} \tag{23}$$

여기서, m 은 로봇의 질량, T_l , T_r 는 좌측과 우측의 구동력, L 은 좌측 또는 우측 구동바퀴와 로봇의 무게중심 사이의 거리, θ 는 로봇의 방향각, V 는 로봇의 속도, M_V 는 로봇의 무게중심에 대한 관성 모멘트이다.

오른쪽과 왼쪽 구동바퀴에 대한 구동 시스템의 동특성은 다음 식으로 나타낸다.

$$\begin{aligned}
 M_w \ddot{\phi}_i + f_c \dot{\phi}_i + RT_i - f_k \Gamma_i &= 0 \\
 i &= r, l
 \end{aligned} \tag{24}$$

여기서, M_w 는 바퀴의 회전 관성 모멘트, f_c 는 점성 마찰계수, f_k 는 구름 이득 계수, R 은 바퀴의 반경, ϕ_i 는 바퀴의 회전각, Γ_i 는 구동 입력 그리고, 변수 θ, V, ϕ_i 사이의 기하학적인 관계는 다음과 같이 주어진다.

$$R\dot{\phi}_r = (V + L\dot{\theta}) \quad (25-a)$$

$$R\dot{\phi}_l = (V - L\dot{\theta}) \quad (25-b)$$

이들 방정식으로 부터, $x = [V \ \theta \ \dot{\theta}]^T$ 를 이동형 로봇에 대한 입력 상태변수, $\Gamma = [\Gamma_r \ \Gamma_l]^T$ 를 제어 입력변수, $y = [V \ \theta]^T$ 를 출력변수로 정의하면, 상태방정식은 다음 식과 같다.

$$\dot{x} = \alpha x + \beta \Gamma \quad (26)$$

$$y = \gamma x \quad (27)$$

여기서,

$\alpha =$

$$\begin{bmatrix} -2f_c(mR^2 + 2M_w)^{-1} & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -2f_c L^2 (M_v R^2 + 2M_w L^2)^{-1} \end{bmatrix}$$

$\beta =$

$$\begin{bmatrix} f_k R (mR^2 + 2M_w)^{-1} & f_k R (mR^2 + 2M_w)^{-1} \\ 0 & 0 \\ f_k R L (M_v R^2 + 2M_w L^2)^{-1} & -f_k R L (M_v R^2 + 2M_w L^2)^{-1} \end{bmatrix}$$

$$\gamma = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

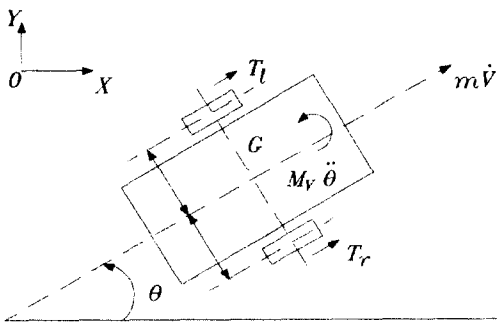


Fig. 5 The structure of a mobile robot

3.2 시뮬레이션

이동형 로봇의 성능시험은 뉴턴-퍼지제어기법에 의해 컴퓨터 시뮬레이션을 통하여 수행하였으며, 미지의 외란 및 파라미터의 불확실성을 고려한

상태에서 속도 및 방향 제어에 대한 성능 시험결과를 토크 계산법과 비교하였다. 시뮬레이션 조건은 적분 구간 1[ms]를 가진 4차 Runge-Kutta method를 이용하였으며, 샘플링 기간은 5[ms]인 것으로 가정하고, 기준속도 V_d 는 300[mm/s]로서 주어졌고, 상태변수의 초기값은 $x = [0 \ 0 \ 0]^T$ 로 한다. 파라미터들은 추정한 상태에서 7개의 제어라벨들이 각각 입력변수에 적용된 49개의 제어규칙을 사용하였으며, 상수들은 각각의 뉴턴-퍼지에 대한 연결 가중치 Ω_b 로서 적용하였다. 중심값 Ω_c 는 -9, -6, -3, 0, 3, 6, 9 이고, 편차 Ω_d 의 보상값들은 후원집합 [-9, 9]상의 모든 라벨들을 똑같이 분배하기 위한 단일체이다.

이동형 로봇의 파라미터들의 사양은 다음과 같다.

$$M_v = 10[\text{kg} \cdot \text{m}^2], \quad m = 200[\text{kg}], \quad L = 0.3[\text{m}], \\ f_k = 5, \quad M_w = 0.005[\text{kg} \cdot \text{m}^2], \quad f_c = 0.05 [\text{kg/s}], \\ R = 0.1[\text{m}].$$

뉴턴-퍼지에 대한 연결 가중치 Ω_s 와 Ω_b 는 앞 절에서 언급한 연결 가중치 1.0, 1.0, 1.0, -1.0 으로서 고정되었다는 가정하에서 학습하였으므로, Ω_c 와 Ω_d 의 학습비는 모두 0이다.

입력계수 Ω_s 에 대한 가중치들의 초기값은 3.0, 3.0, 4.5, 4.5으로 고정하였고, 학습비는 $\xi = 2.5 \times 10^{-3}$ 으로 하였다. Ω_b 의 학습비는 속도와 방향각에 대하여 각각 정의하였다. 즉 속도와 관련된 뉴턴-퍼지의 속도 학습비는 $\xi_v = 3.1 \times 10^{-4}$ 이고, 방향각과 관련된 뉴턴-퍼지의 방향각 학습비는 $\xi_\theta = 7.71 \times 10^{-5}$ 이다. 원형 궤적을 조종하기 전에 이동형 로봇의 초기 속도가 0이기 때문에 희망한 속도 300[mm/s]에 도달하도록 가속하여야 한다.

Ω_s 의 관련 학습결과들은 Table 1에 도표화하였고, 시뮬레이션 후, Table 2에 나타난 초기값들에 의하여 Ω_b 를 얻었다.

Table 1 The learning results of Ω_s for reference trajectory

		초기치		4차반복	
Ω_s	속도(V)	300	300	300.18	300.11
	방위(θ)	257.8	257.8	255	257.9

Table 2 The 49 control rules for 7 lables

δ	δ						
	NB	NM	NS	ZO	PS	PM	PB
NB	-9	-9	-6	-9	-6	-6	-6
NM	-9	-6	-3	-6	-6	-6	-3
NS	-6	-3	-3	-3	0	3	6
ZO	-9	-6	-3	0	3	6	9
PS	-6	-3	0	3	3	3	6
PM	3	6	6	6	3	6	9
PB	6	6	6	9	6	9	9

Fig. 6은 10%의 불확실성 파라미터를 고려한 상태에서 제안된 뉴럴-퍼지제어기 속도궤적의 추적 성능에 대한 실험 결과이다. Fig. 6-(a)는 두가지의 속도 기준궤적의 상태를 나타내고, Fig. 6-(b)는 기준궤적1에 대한 추종오차이고, Fig. 6-(c)는 기준궤적2에 대한 추종오차이다. 그리고 Fig. 6-(d)는 기준궤적의 추종에 대한 궤적 추종 응답 상태를 나타낸다. 성능실험 결과 10% 파라미터 불

Table 3 The experimental results for the performance of tracking of NF velocity and trajectory of NF controller

파라미터 불확실성	최대오버슈트량	정착시간
20%	0.5[m/s]	0.1초

Table 4 The experimental results for the performance of tracking of velocity and trajectory of computer aided torque controller

파라미터 불확실성	최대오버슈트량	정착시간
10%	1.5[m/s]	0.4초

Table 5 The experimental results for the performance of tracking of direction angle and trajectory of NF controller

파라미터 불확실성	최대오버슈트량	정착시간
10%	0.007[rad]	0.3초
20%	0.008[rad]	0.3초

확실성시 최대 오버슈트량은 약 0.35[m/s]를 나타내고, 정착시간(Settling time)은 약 0.1초를 나타내고 있으므로, 두가지 속도 기준궤적에 대한 추적 성능이 매우 양호함을 보여주고 있다.

Table 6 The experimental results for the performance of tracking of direction angle and trajectory of computer aided torque controller

파라미터 불확실성	최대오버슈트량	정착시간
10%	0.025[rad]	1.2초

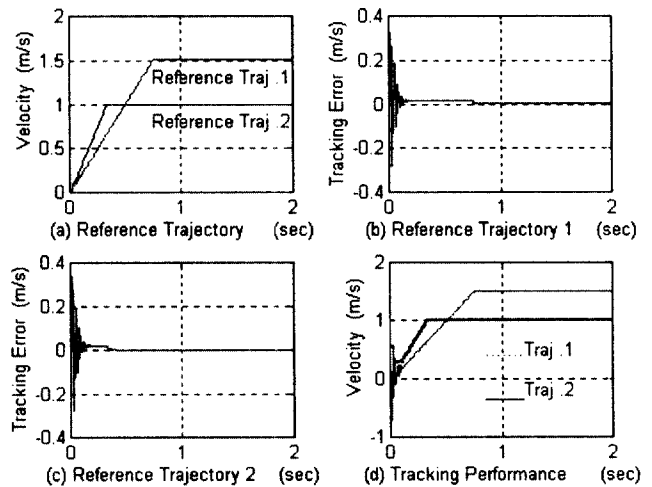


Fig. 6 The control performance of NF controller for the variation of the velocity and trajectory with parameter uncertainties(10%)

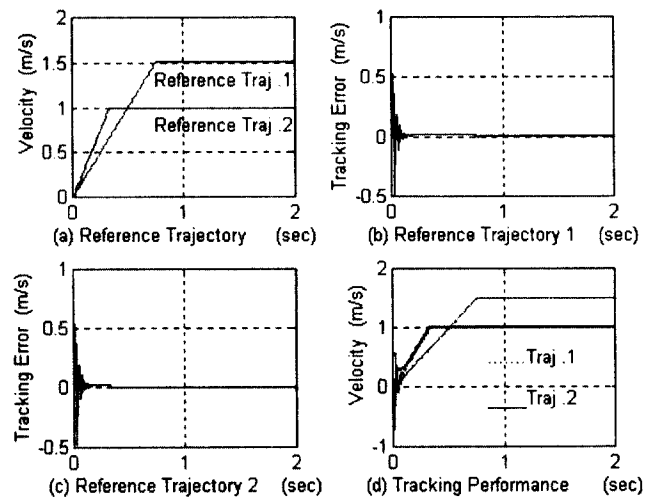


Fig. 7 The control performance of NF controller for variation of the velocity and trajectory with parameter uncertainties(20%)

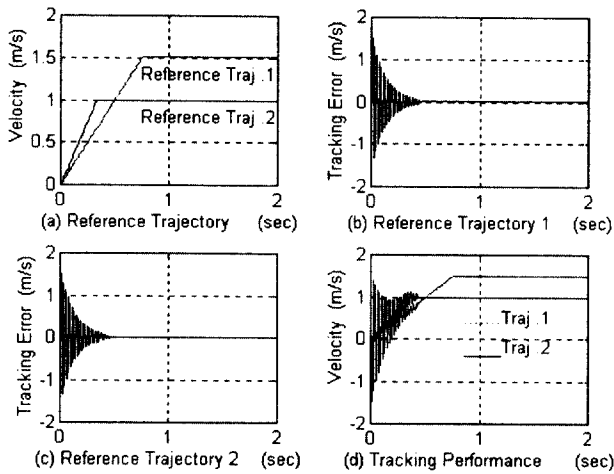


Fig. 8 The control performance of computer torque method for the variation of velocity and trajectory with parameter uncertainties (10%)

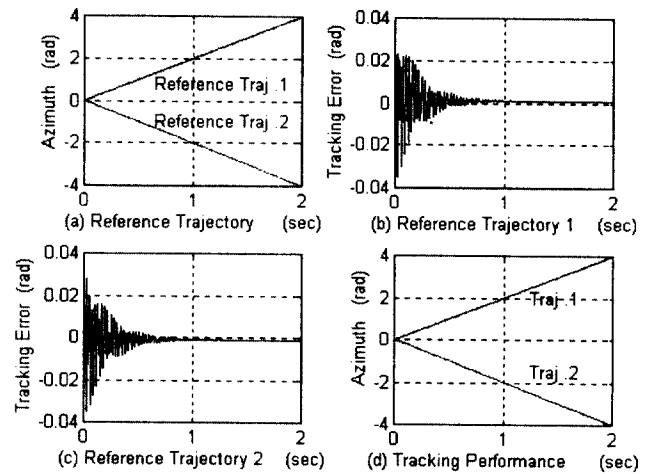


Fig. 11 The control performance of computer torque method for the variation of azimuth and trajectory with parameter uncertainties (10%)

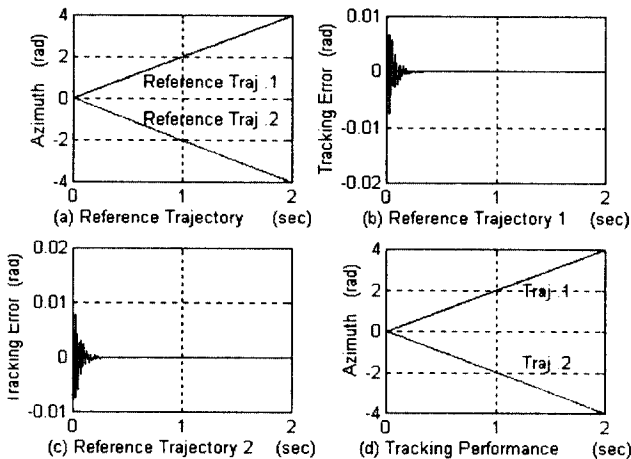


Fig. 9 The control performance of NF controller for the variation of azimuth and trajectory with parameter uncertainties (10%)

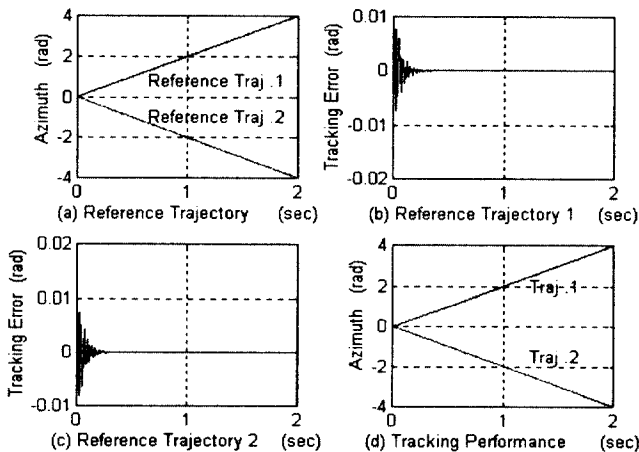


Fig. 10 The control performance of NF controller for the variation of azimuth and trajectory with parameter uncertainties (20%)

4. 결론

본 연구에서는 복잡한 동특성을 지닌 불확실성 시스템의 견실 정밀제어를 위해 시스템 자체에 대한 사전 정보 없이 변화하는 제어 환경에 대하여 학습능력에 의해 스스로 적응할 수 있는 뉴럴제어 구조와 퍼지 추론을 통해 제어입력을 생성하는 퍼지제어 구조를 결합한 새로운 구조의 실시간 제어 가능한 뉴럴-퍼지제어기법을 제안하였다.

제어구조의 특징은 가우스 멤버쉽함수를 적용한 퍼지추론법과 역전파 알고리즘을 통한 다층구조의 뉴럴 제어구조로 설계함으로써 미지의 외란 및 파라미터의 불확실성에 대한 매우 향상된 견실성이 유지되는 제어구조라 할 수 있다.

제안된 뉴럴-퍼지제어기의 성능을 확인하기 위하여 시스템의 정확한 동적 모델링이 어려운 불확실성 시스템인 이동형 로봇의 속도 및 방향각 제어에 대하여 시뮬레이션을 통하여 매우 양호함을 예증하였다.

참고문헌

1. Hemami A., Mehrabi M. G. and Cheng R. M. H., "Synthesis of an Optimal Control Law for Path Tracking in Mobile Robots," *Automatica*, Vol. 28, No. 2, pp. 383~387, 1992
2. Iwata T., Machida K. and Toda Y., "Fuzzy Control Using Neural Network Techniques,"

- World Congress on Neural Networks, Vol. 3, pp. 365~370, 1993
3. Mamdani E. H., "Application of fuzzy algorithms for control of simple dynamic plant," Proc. IEEE, Vol. 121, No. 12, pp. 1585~1588, 1974
 4. Rumelhart D. E. and McClelland J. L., "Parallel Distributed Processing, Explorations in the Microstructure of Cognition," MIT Press, Vol. 1, Mass., 1987
 5. Horikawa S., Furuhashi T., Okumas S. and Uchikawa Y., "A Learning Fuzzy Controller Using a Neural Network," Trans. SICE, Vol. 27, No. 2, pp. 208~215, 1991
 6. Horikawa S. etc., "A Fuzzy Controller Using a Neural Network and Its Capability to Learn Expert's Control Rules," IIZUKA'90, pp. 103~106, 1990
 7. Sugeno M., "Fuzzy Control," Nikkan-kogyo shinbunsha, Tokyo, 1988
 8. Psaltis D., Sideris A. and Yamamura A., "A Multilayered Neural Network Controller," IEEE Control Systems Magazine, pp. 17~21, 1988
 9. Mizumoto M., "Simple Fuzzy Theories," Computrol, Vol. 28, pp. 32~45, 1989
 10. Mizumoto M., "Fuzzy Reasoning Methods for Fuzzy Control," J. of the Society of Instrument and Control Engineers, Vol. 28, No. 11, pp. 959~963, 1989
 11. Mamdani E. H., "Application of fuzzy algorithms for control of simple dynamic plant," Proc. IEEE, Vol. 121, No. 12, pp. 1585~1588, 1974
 12. Rumelhart D. E. and J. L. McClelland, "Parallel Distributed Processing, Explorations in the Microstructures of Cognition," MIT Press, Vol. 1, Foundations, 1986
 13. Watanabe .K. and Tzafestas S. G., "Learning Algorithms for Neural Networks with the Kalman Filters," J. Intelligent and Robotic Systems, Vol. 3, No. 4, pp. 305~319, 1990
 14. Craig J. J., "Introduction to Robotics, Addison-Wesley," 1989