

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

한 성 현*
(96년 2월 25일 접수)

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

Sung-Hyun Han*

Key Words : Azimuth(방향각), Back Propagation Algorithm(역전파 알고리즘), Neural-Fuzzy Control(뉴럴-퍼지제어), Membership Function(멤버십함수), 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 wheel. The proposed control scheme uses a Gaussian function as a unit function in the fuzzy-neural network, and back propagation algorithm to train the fuzzy-neural network controller in the framework of the specialized learning architecture. It is proposed a learning controller consisting of two neural network-fuzzy based on independent reasoning and a connection net with fixed weights to simply the neural networks-fuzzy. The performance of the proposed controller is shown by performing the computer simulation for trajectory tracking of the speed and azimuth of a mobile robot driven by two independent wheels.

1. 머리말

기존의 이동 로봇의 제어기는 제어하려는 물체의 운동방정식을 알고 있을 때만 가능하다. 그러나 이동로봇의 구조는 이동로봇의 운동방정식이 비선형성을 나타냄으로써 복잡하고 대부분의

경우 운동방정식을 정확히 구하지 못하는 경우가 많다¹⁾. 또한 널리 사용되는 고전적 제어의 경우 이득값이 고정되어 있다. 이러한 고전적 제어방법은 외부 환경의 변화에 대응할 수 없으므로 제어성능에 한계점을 드러내고 있다. 최근에 뉴럴-퍼지(neural network-fuzzy)이론을 제어시스템에 적

* 경남대학교 공과대학 기계설계학과

용함으로써 고전적 제어이론의 한계성 극복에 커다란 가능성을 보이고 있다.

뉴럴-퍼지제어 방법은 불확실성 시스템의 특성을 지니는 이동용 로봇에 대한 정확한 수학적인 동적 모델링이 요구되지 않는 것이 특징이라 할 수 있으므로²⁾, 선형시스템은 물론 비선형시스템에도 쉽게 적용할 수 있는 장점을 지니고 있기 때문에 파라미터의 동특성이 복잡한 비선형시스템을 제어하기에 적절한 제어기법³⁾으로서 국내외적으로 많은 연구가 진행되고 있다. 퍼지제어이론을 이용한 퍼지제어기의 설계시에는 퍼지제어기의 입력 데이터와 출력 데이터, 멤버쉽함수의 수, 멤버쉽함수의 범위에 관하여 제어기의 파라미터들을 맞추어야 하고, 신경회로망 이론을 사용한 신경망 제어는 기율기법 또는 역전파 알고리즘을 이용하여 모델링과 추정을 할 수 있는 특성을 지니며, 집합적 처리능력은 복잡한 지각입력들에 대한 빠른 응답을 제공해 주는 것이 특징이다. 신경망 제어기법의 특징은 입력과 출력정보가 불확실할지라도 신경망 학습기능으로 인하여 좋은 역동력학적 효과를 얻을 수 있다. 신경망은 비선형성, 학습, 병렬처리⁴⁾ 등의 장점을 가지며, 신경망 학습법에서 비지도 학습은 외부교시에 의존하지 않지만 교시가 학습법으로 고려될 수 있고, 지도 학습은 외부교시를 요구하고 입력에 대한 희망출력을 산출하는 학습시스템을 제공하며, 보강학습은 교시라기 보다는 추정하는 것에 가깝고, 입력에 대한 정확한 출력을 발견하고 학습의 출력을 기억한다.

신경망 학습과 퍼지이론을 융합시키는 기술은 상당한 관심사로 부각되고 있다. 신경-퍼지제어기의 기본원리는 신경단위 세포의 학습함수를 이용한 퍼지제어칙과 멤버쉽함수를 자동적으로 조정하는 제어방법이라 할 수 있다.

전문가에 의하여 구동되는 플랜트로부터 처리된 패턴 데이터로써 제어칙을 확인할 수 있고, 그리고 선행단과 끝단이 관련된 파라미터들을 정확히 맞춘다⁶⁾. 이에 대표적인 접근 방법인 호리카와에 의한 방법은 일반화된 학습 구조가⁷⁾ 신경망을 다루는데 사용되었기 때문에 제어데이터의 형태가 없

는 경우에는 적용될 수 없는 특성이 있다. 다시 말하면, 이 제어구조의 접근 방법은 유사 사다리꼴류 함수가 다른 기호를 가진 두개의 시그모이드 단위 함수(sigmoidal unit functions)를 합한 것에 의해서 구성되기 때문에 선행단에서의 멤버쉽함수를 산출하기 위한 중간층(intermediate layer)이 요구되는 방법으로서 이와 같은 방법을 퍼지 시그모이드 신경망이라고 부른다. 또한, 연결 중간층에서 단일체(unity)의 수는 퍼지 표시들의 수가 증가함으로써 성장한다. 또한, 이동형 로봇의 개발은 앞으로 무한한 개발가능 영역인 해저탐사 및 해저 개발을 위해서도 매우 중요한 역할을 차지하리라 생각된다. 본 연구에서는 이동용 로봇의 속도 및 방향제어를 위하여 역전파알고리즘에 의한 다층신경망 구조 및 가우스 멤버쉽함수의 혼합 퍼지 추론법을 적용한 새로운 구조의 뉴럴-퍼지 (Neural-Fuzzy, NF)제어기법이 제안되고, 제안된 뉴럴-퍼지제어기법에 의해 설계된 이동형 로봇 제어기의 성능이 컴퓨터 시뮬레이션을 통하여 예증된다.

2. NF 제어기 설계

2.1 추론방법 및 멤버쉽함수

본 연구에서 채택된 퍼지제어에서 퍼지추론은 수지노의 퍼지추론⁸⁾의 특별한 경우로써 설명될 수 있는 단순화된 혼합 추론법이다. 사실, 퍼지 신경회로 제어법은 수지노의 퍼지추론의 끝단의 함수가 상수가 되거나 또는 최소-최대 도심법의 끝단에서의 퍼지집합의 범위가 미소값, 즉 싱글톤(singleton) 이 될 때의 경우와 일치한다. 따라서, 임의의 i 번째 제어칙은 다음과 같이 쓸 수 있다.

$$\begin{aligned} H_i : & \text{If } x_1 = M_{i1} \text{ and } \dots \text{ and } x_n = M_{in} \\ & \text{then } \Gamma_1 = N_{i1} \text{ and } \dots \text{ and } \Gamma_n = N_{in} \end{aligned} \quad (1)$$

여기서, H_i 는 i 번째 제어칙을 나타내고, M_{ij} 는 i 번째 제어칙에서 j 번째 입력변수와 결합되는 선행

단에서의 퍼지집합이고 N_{ij} 는 i 번째 제어칙의 끝단에서의 j 번째 변수와 결합되는 상수를 나타낸다. 신뢰도 $\sigma M_{i1}(x_1), \dots, \sigma M_{im}(x_m)$ 을 적용하면, 선행단에서의 신뢰도 Q_i 는 다음과 같이 정의된다.

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

j 번째 출력 결과는 가중치 Q_i 에 관하여 N_{ij} 의 가중 평균치로써 다음과 같이 계산된다.

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

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

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

여기서, μ 는 영보다 작은 상수이다.

Fig. 1은 신경망을 이용한 구조를 보여준다. 여기서 기호 $\{ \}$ 를 가진 변수는 신경망을 통한 신호의 통과를 나타낸다. 원형기호는 단일체(unity)이고, Ω_s 는 입력변수 x_i 에 대한 계수이고 Ω_c 와 Ω_g 는 각각 중심(즉, 함수값 0.5와 일치하는 비례 후원집합에 관한 값)과 시그모이드 함수의 구배를 나타내는 연결 가중치들이다. 덧붙여서, 기호 -1을 가지는 단일체(unity)는 -1의 결과를 산출하고 Σ 기호를 가지는 단일체(unity)는 입력들의 합을 산출한다. 그리고 기호 F를 가지는 단일체(unity)에서의 입력과 출력의 관계는 식(4)에 의해서 정의된다. 또한 기호를 가지지 않는 단일체(unity)는 입력을 출력으로 간단히 산출한다. Fig. 1에서 보여지는 신경망을 적용하면, 영에서 중심을 가지는 결과류 함수는 식(5)와 식(5)의 음의 값을 부과함으로써 구성될 수 있다.

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

여기서, a, b, c 및 d는 임의의 상수를 나타낸다.

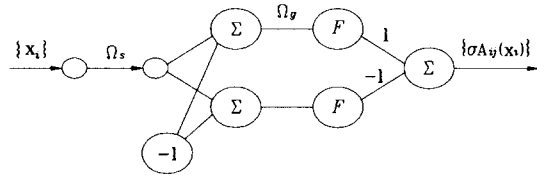
퍼지 시그모이드 신경회로망은 유사 사다리꼴류 함수가 다른 기호를 가진 두개의 시그모이드

유니트 함수를 합하는 것으로 구성되기 때문에, 선행단에서의 멤버쉽함수를 산출하기 위한 부과의 중간층들을 요구한다는 것을 생각할 필요가 있다. 또한 이러한 사실은, 연결 중간층에서의 단일체(unity)들의 수는 퍼지 라벨들의 수가 커짐으로써 성장하는데 원인이 있다. 퍼지-신경망법은 단위 함수로써 정의되는 가우스 함수로써 정의할 수 있다.

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

이 경우에 대하여, 신경망을 이용한 멤버쉽 함수의 구성은 Fig. 2에 나타내고 있다. 여기서, 연결 가중치 Ω_c 는 가우스 함수에 대한 중심값을 나타내고, Ω_g 는 표준화된 후원집합에서 가우스함수 0.5를 갖는 중심 Ω_c 로부터 편차의 역의 값을 나타낸다. 결과적으로, 신경망과 식(6)의 계산을 이용해서 다음과 같은 가우스류 함수를 나타낼 수 있다.

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



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

Fig. 1 Neural network for generating a membership function

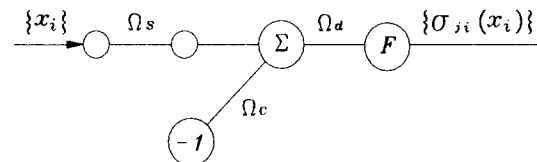


Fig. 2 Neural network for gaussian membership function

2.2 NF 제어기의 구조 및 원리

Fig. 3는 두개의 입력 (x_1, x_2)와 하나의 출력 (Γ_1^*)과 선행단에서의 세개의 라벨들이 있는 경우에 대하여 퍼지-신경망의 구성 예를 든다. 그때, 증명할 수 있는 제어칙들의 수는 $r=3^2$ 이다. 그림의 (VI)층에 대하여 Σ 와 P의 기호를 갖는 단일체(unity)는 선형 덧셈 입력을 갖는 같은 결과를 산출한다.

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

아래의 Fig. 3에서 (I) ~ (V)층은 퍼지제어칙의 선행단에 해당하고 (VII)과 (VIII)층은 끝단에 해당한다. (I)층에 적용된 입력들 x_i 는 연결가중치 Ω_s 와 비례된다. (III)층에서 기저(Bias) 인 연결가중치 -1 는 (IV)층에서 가우스 함수에 대한 입력인 Ω_d 에 의해 곱해졌다. (V)층에서는 모든 제어칙들에 대한 선행단에서 신뢰도 Q_i 를 얻는다. (VI)층의 첫번째 단일체(unity)에서 입력들의 합과 역 계산을 수행한다. 끝으로, (VII)층과 (VIII)층에서 가중치 Q_i 에 관한 Ω_b 의 가중평균치

로써 결과값을 얻는다. 여기서, Ω_b 는 식(10)에서 정의된 N_j 에 대한 실제 연결가중치 값을 내고 있다.

역전파알고리즘을 적용함으로써 뉴럴-퍼지제어기의 연결 가중치들을 학습시킬 수 있고, 따라서 제어칙들을 검증할 수 있고 선행단에서 멤버십함수들을 정확히 맞출 수 있다. 다음에서, 플랜트의 결과 편차를 전문가에 의하여 산출된 패턴 데이터를 이용하지 않고 최소화시키기 위하여 퍼지-신경망을 학습시키는 특별한 학습구조에 기초를 둔 경우를 설명하기로 한다. 그렇지만, 단지 일반화된 학습구조 또는 피드백 오차 학습구조에 기초를 둔 경우에 대하여 결과층에서 단지 η 값을 변화시킨다. 식(10)은 K번째층에 대한 입력이 함수 F(\cdot)를 통한 출력이고, (K+1)번째 층에 대한 입력이 덧셈(즉 Σ)작업에 의해 계산될 때의 경우로 나타낸다고 하자. 유사하게, 식(11)은 K번째 층에 대한 입력이 함수 F(\cdot)를 통한 출력이고, (K+1)번째층에 대한 입력이 대수적에 의해서 계산되는 경우로 표시된다고 하면, 출력층 q에서의 j번째 단일체내의 η_j^q 와 어떤 중간층 K에서 j번째 단일체내의 η_j^K 는 각각

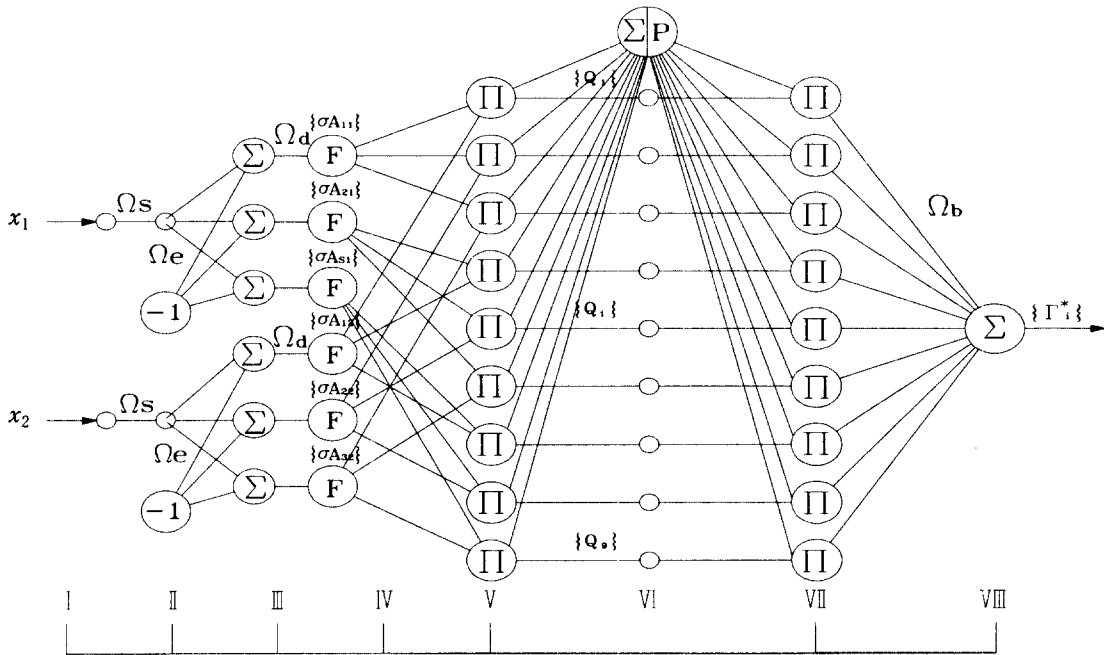


Fig. 3 Basic structure of neural-fuzzy network

식(9), 식(10) 및 (11)로 주어진다. 여기서 η 는 원하는 출력 (x_{di})과 실제출력 (x_i)과의 오차 활성치 함수의 미분량 및 구동입력의 출력에 대한 Jacobian으로 정의되는 양이다.

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

$$\eta_j^k = F'(I_j^k) \sum_j \eta_i^{k+1} \Omega_{ij}^{k,k+1} \quad (10)$$

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

여기서, 식(9)은 출력층 q에 대한 η 값을 나타내고, 식(9),(10)은 중간층에 대한 η 의 값을 계산하고 있다. 그러므로, η 값의 계산에 대하여 (10)의 적용은 (IV)와(VI)층에서 계산된다. 여기서, (IV)와(VI)층의 첫번째 단일체(unity)에서의 F' 을 고려하면, 다음식과 같이 정의된다.

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

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

식(12)는 (IV)층에 대한 값이고, 식(13)은 (VI)층의 첫번째 단일체(unity)에 대한 값이다. 그리고 다른 선형 단일체(unity)에 대하여 $F'(I_j^k) = 1$ 로 계산된다.

또한 $\frac{\partial x_i(kT)}{\partial \Gamma_i(kT)} \approx \frac{\Delta x_i(kT)}{\Delta \Gamma_i(kT)}$ 로서 식(10)의 자코비안 $\frac{\partial x_i}{\partial \Gamma_j}$ 의 값을 구한다. 여기서, $\Delta \Gamma_i(\cdot)$ 와 $\Delta x_i(\cdot)$ 는 입력과 출력데이터 $\Delta = 1 - d^{-1}$ 로부터 산출된다. 그리고 d^{-1} 는 한 스텝지연 동작이고, K는 이산시간이고 T는 샘플링 범위이다. 만약 플랜트가 입력의 어떤 시간 지연을 갖지 않는 최초의 이산시간 지연 시스템이라면 위의 방정식 대신 $\frac{\partial x_i(kT)}{\partial \Gamma_i[(k-1)T]}$ 값을 의미한다.

위의 결과들은 연결 가중치들의 갱신된 방정식 (14),(15)를 산출한다.

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

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

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

식(14)는 식(12), 식(15)는 식(13)에 대한 각각의 값이다. 여기서, t는 t번째 갱신시간을 나타내고 ξ 는 학습비를 의미하는 작은 양의 상수이고, $\Delta \Omega_{ij}^{k-1,k}(t)$ 는 t번째 구간에서 연결 가중치의 증분이고, b는 안정요소를 의미하는 작은 양의 상수이다. 그러므로, 식(14)을 이용하여 연결 가중치 Ω_s , Ω_c , Ω_d , Ω_b 을 갱신할 수 있다. 여기서, 식(15)와 관련된 연결 가중치들이 모두 단일체(unity)로써 고정되었기 때문에, 식(15)에 의한 어떤 연결 가중치들의 갱신방정식을 사용하지 않는다. 그리고 입력층에서 두입력신호 y_1 및 y_2 가 서로 매우 다른 경우에도 그에 따른 척도요소(scaling factor) Ω_s 에도 또한 매우 심하게 달라지게 된다. 이러한 경우에는 보통 척도요소를 두 종류로 분리하여 설정하여야만 되므로 이러한 경우 그 문제점을 해결하기 위하여 다음과 같이 수정된 연결가중치 Ω_s 의 갱신방정식 식(16)을 제안한다.

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

위의 식(14)에서 η_j^k 와 z_i^{k-1} 은 각각 $\Omega_{ij}^{k-1,k}(t)$ 으로부터 일반화되었음을 나타낸다.

2.3 NF 제어기의 학습구조

본 논문에서는 구동토크와 플랜트의 출력 정보 사이의 관계식을 얻기 위하여 제어기의 영역에서 폭넓게 사용되는 컴퓨터 토크제어기를 모델기준서보제어기로 활용하였다. 이러한 설계기법으로부터 얻어진 제어원리 및 기본구조가 2개의 퍼지-신경망으로 구성되는 학습제어기에 적용된다. 이동형 로봇의 동적 모델링식으로 부터 다음 식으로 표현될 수 있다.⁶⁾

$$\begin{aligned} \Gamma_v &= \frac{\theta^*}{f_k R L (M_v R^2 + 2M_w L^2)^{-1}} \\ &- \frac{-2f_c L^2 (M_v R^2 + 2M_w L^2)^{-1} \theta}{f_k R L (M_v R^2 + 2M_w L^2)^{-1}} \end{aligned}$$

$$+ \frac{\dot{V}^*}{f_k R(mR^2 + 2M_w)^{-1}} - \frac{-2f_c V(mR^2 + 2M_w)^{-1}}{f_k R(mR^2 + 2M_w)^{-1}} \quad (17)$$

$$\Gamma_l = - \frac{\ddot{\theta}^*}{f_k RL(M_v R^2 + 2M_w L^2)^{-1}} - \frac{-2f_c L^2(M_v R^2 + 2M_w L^2)^{-1} \dot{\theta}}{f_k RL(M_v R^2 + 2M_w L^2)^{-1}} + \frac{\dot{V}^*}{f_k R(mR^2 + 2M_w)^{-1}} - \frac{-2f_c V(mR^2 + 2M_w)^{-1}}{f_k R(mR^2 + 2M_w)^{-1}} \quad (18)$$

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

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

$$\dot{V}^* = \dot{V}_d + K_{VP} \delta_V \quad (20)$$

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

$$\delta_\theta = \theta_d - \theta \quad (21)$$

$$\delta_V = V_d - V \quad (22)$$

여기서, θ_d 와 V_d 는 각각 방향각과 기준속도들로 정의한다. 로봇트에 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.

로봇트의 속도와 방향각은 왼쪽과 오른쪽 구동바퀴들에 대한 토크를 조정함으로써 제어된다. 즉, 여기서 고려된 제어시스템은 다입력 다출력이다. 입력으로써 $\delta_V, \delta_\theta, \dot{\delta}_V, \dot{\delta}_\theta$ 와 추론 출력으로써 Γ_r, Γ_l 을 갖는 퍼지-신경망에서, 선행단의 제어

칙의 수는 선행단에서 라벨 l 의 수가 증가함으로써 l^4 으로 성장한다. 이것은 신경망에서의 중간 단 일체(unity)들의 수의 지수함수적인 증가에 원인이다.

제안된 제어기에서 모델기준 서보제어의 구조를 분석하면, $\Gamma_r = \Gamma_V + \Gamma_\theta$, $\Gamma_l = \Gamma_V - \Gamma_\theta$ 의 관계식이 주어짐을 알 수 있다. 여기서, Γ_V 는 로봇트의 속도의 측정을 이용하여 로봇트의 속도를 제어하기 위하여 요구되는 토크이고, Γ_θ 는 로봇트의 방향각의 측정을 이용하여 로봇트의 방향각을 제어하기 위하여 요구되는 토크이다. 만약 Γ_V 가 입력으로써 속도오차 δ_V 와 속도오차의 비를 가진 퍼지신경망(FNN₍₁₎)과, Γ_θ 가 입력으로써 방향각 오차 δ_θ 와 방향오차의 비를 갖는 퍼지신경망(FNN₍₂₎)으로부터 산출된다면, 로봇트의 속도와 방향각에 대한 학습제어기는 구성된다. 이러한 사실로부터 선행단에서 연결 가중치 $\Omega_s, \Omega_c, \Omega_d$ 들에 의하여 각각의 퍼지신경망에 대한 멤버십함수를 정확히 맞출 수 있고, 그리고 상수들 Ω_b 에 의하여 제어칙들을 확인할 수 있다.

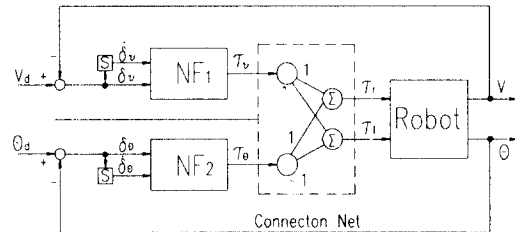


Fig. 4 The global structure of neural-fuzzy controller

3. 시뮬레이션

3.1 이동형 로봇트의 동적 모델링

두개의 독립 구동바퀴를 가지는 이동 로봇트가 평면상에서 정확히 움직이며, 절대 좌표계 O-XY는 평면상에 고정되었다고 가정한 상태에서 이동 로봇트의 운동방정식은 다음과 같이 표현된다.

$$M_V \ddot{\theta} + (-T_r + T_l)L = 0$$

$$m\dot{V} - (T_l + T_r) = 0 \quad (23)$$

오른쪽과 왼쪽 구동바퀴에 대한 구동 시스템의 동특성은 다음의 식으로 표현된다.

$$M_w \ddot{\phi}_i + f_c \dot{\phi}_i + RT_i - f_k \Gamma_i = 0, \\ i = r, l \quad (24)$$

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_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, M_V 는 로봇의 무게중심에 대한 관성 모멘트 또한, 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 방정식들로부터, $y = [V \ \theta]^T$ 를 로봇에 대한 입력 상태변수, $\Gamma = [\Gamma_r \ \Gamma_l]^T$ 를 제어 입력 변수, 그리고 $x = [V \ \theta]^T$ 를 출력변수로 정의하면, 상태방정식은 다음의 식(26)과 식(27)으로 될 수 있다.

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

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

여기서,

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

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

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

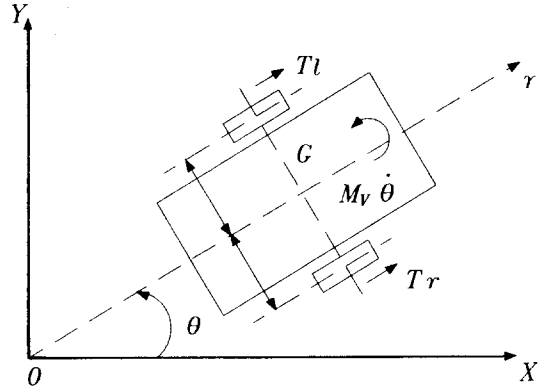


Fig. 5 The structure of a mobile robot

3.2 실험

이동형 로봇의 성능시험을 NF제어기법에 의해 컴퓨터 시뮬레이션을 통하여 수행하였다. 수치 실험은 미지의 외란 및 파라미터의 불확실성을 고려한 상태에서 이동용 로봇의 속도 및 방향 제어에 대한 성능 시험을 수행하였고 그 결과를 토크 계산법과 비교하였다. 위의 이동 로봇 모델의 시뮬레이션 조건은 적분 구간 1[ms]를 가진 4차 Runge-Kutta 법을 이용하였고, 샘플링 기간은 5[ms]인 것으로 가정한다. 또한, 기준속도 V_d 는 300[mm/s]로써 주어졌고, 상태변수의 초기값은 $x = [0 \ 0 \ 0]^T$ 로써 주어졌다. 로봇의 물리적인 파라미터들은 완전하게 알려지지 않은 것으로 추정된 상태에서 Table 2.에서 보여진 것처럼 7개의 제어 라벨들이 각각 입력변수에 적용된 49개의 제어척을 사용했다. 즉 이 표에서 상수값들은 각각의 퍼지-신경망에 대한 초기 연결 가중치들 Ω_b 로써 적용되었다. 7개의 라벨들 중 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 위하여 모두 단일체(unity)에서 이다. 이동 로봇의 물리적인 파라미터들의 사양은 다음과 같다. $M_V = 10[\text{kg m}^2]$, $m = 200[\text{kg}]$, $L = 0.3[\text{m}]$, $f_k = 5$, $M_w = 0.005[\text{kg m}^2]$, $f_c = 0.05 [\text{kg/s}]$, $R = 0.1[\text{m}]$. 본 시뮬레이션에서 각각의 퍼지-신경망에 대한 연결망의 가중치 Ω_s 와 Ω_b 는 앞절에서 토론된 것처럼 연결망의 가중치들은 1.0, 1.0, 1.0, -1.0

로써 고정되었다는 가정하에서 학습되었다. 여기서, 퍼지-신경망의 다른 연결 가중치들은 학습되지 않았다. 즉 Ω_c 와 Ω_d 의 학습비는 모두 0으로써 고정되었다.

입력계수 Ω_s 에 대한 가중치들의 초기값은 3.0, 3.0, 4.5, 4.5으로 고정되었고, 학습비는 $\xi=2.5 \times 10^{-3}$ 로 정의되었다. Ω_b 의 학습비는 속도와 방향각에 대하여 각각 정의되었다. 즉 속도와 관련된 퍼지-신경망의 속도 학습비는 $\xi_v=3.1 \times 10^{-4}$ 이고, 방향각과 관련된 퍼지-신경망의 방향각의 학습비는 $\xi_\theta=7.71 \times 10^{-5}$ 으로 정의되었다. 원형 궤적을 조종하기 전에 이동로봇의 초기 속도가 0[m/s] 이기 때문에 로봇의 속도가 희망 속도 300[mm/s]에 도달하도록 로봇을 가속해야 한다. 이러한 관점으로 부터, 단지 속도제어가 도달된 워밍업 시간을 소개하고, 반면 방향각은 초기 상태로 유지한다. 여기서, 속도제어가 도달된 워밍업 시간을 T_w 로 나타낸다. 한번의 학습의 시도는 워밍업 시간을 제외한 제어간격이 35[s] 이내로 되었고, 연결 가중치들은 모든 각각의 샘플링 시간마다 갱신되어진다.

Ω_s 의 관련 학습결과들은 Table 1.에서 도표화되었다. 시뮬레이션 실험후, Table 2.에 나타낸 초기값들에 의하여 Ω_b 를 얻었다. 이로부터, 세번의 시뮬레이션 실험후에 만족스런 제어결과들을 얻을 수 있었다. 시뮬레이션 실험에 있어서 시뮬레이션 시 고려된 파라메타의 불확실성의 고려는 주어진 이동형 로봇의 모든 제원에서 휠의 관성모멘트 및 점성마찰력에 대한 파라미터 값을 증감시키는 방법으로 고려하였다.

Fig. 6은 10%의 파라미터의 불확실성을 고려한 상태에서의 제안된 NF제어기의 속도제적추적 성능 실험 결과를 나타내고 있다. 여기서, Fig. 6-(a)는 두가지의 속도 기준제적의 상태를 나타내고, Fig. 6-(b)는 기준제적1에 대한 추종오차를 나타내고, Fig. 6-(c)는 기준제적2에 대한 추종오차를 나타낸다. 그리고 Fig. 6-(d)는 기준제적의 추종에 대한 제적 추종 응답상태를 보여주고 있다. 성능실험 결과 10% 파라미터 불확실성시 최대 오버슈트량은 약 0.35(m/s)를 나타내고, 정착시간(settling time)은 약 0.1초를 나타내고 있다. 두가지 속도기준제적에 대한 추적성능이 매우 양호함을보여주

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 |

고 있다. Fig. 7은 20%의 파라미터 불확실성을 고려한 상태에서 제안된 NF제어기의 속도궤적추적 성능 실험 결과를 나타내고 있다. 파라미터 불확실성을 20% 고려한 상태에서의 제안된 NF제어기의 최대 오버슈트량은 약 0.5(m/s)를 나타내고, 정착시간(settling time)은 약 0.1초를 나타내고 있다. 20%의 파라미터 불확실성을 고려한 상태에서도 제안된 NF제어기는 매우 견실한 제어성능을 유지하고 있다. Fig. 8은 제안된 NF제어기의 성능의 신뢰성을 확인하기 위하여 기존의 컴퓨터-토크 제어기법에 대해 10%의 파라미터 불확실성을 고려한 상태에서의 속도궤적추종 성능실험결과를 나타내고 있다. 기존의 컴퓨터-토크 제어기는 10%의 불확실성을 고려한 상태에서 최대 오버슈트량은 약 1.5(m/s)이며, 정착시간은 약 0.4초를 나타내고 있다. 시뮬레이션 실험결과에서 보는 바와같이 제안된 NF제어기는 기존의 컴퓨터 토크제어기법 보다 파라미터 불확실성에 매우 견실하고 정밀한 제어성능이 유지됨을 확인할 수 있다.

한편, Fig. 9는 10%의 파라미터 불확실성을 고려한 경우에서의 NF제어기의 방향각궤적추적 성능 실험 결과를 나타내고 있다. 여기서, Fig. 9-(a)는 두가지의 방향각 기준궤적의 상태를 나타내고, Fig. 9-(b)는 기준궤적1에 대한 추종오차를 나타내고, Fig. 9-(c)는 기준궤적2에 대한 추

종오차를 나타낸다. 그리고 Fig. 9-(d)는 기준궤적의 추종에 대한 궤적 추종 응답상태를 보여주고 있다. 성능실험결과 10% 파라미터 불확실성 시 최대 오버슈트량은 약 0.007(rad)를 나타내고, 정착시간(settling time)은 약 0.3초를 나타내고 있다. 두가지 방향각기준궤적에 대한 추적성능이 매우 양호함을 보여주고 있다. Fig. 10은 20%의 파라미터 불확실성을 고려한 경우에서 제안된 NF제어기의 방향각궤적추적 성능 실험 결과를 나타내고 있다. 파라미터 불확실성을 20% 고려한 상태에서의 제안된 NF제어기의 최대 오버슈트량은 약 0.008(rad)를 나타내고, 정착시간(settling time)은 약 0.3초를 나타내고 있다. 20%의 파라미터 불확실성을 고려한 상태에서도 제안된 NF제어기는 매우 견실한 제어성능을 유지하고 있다. Fig. 11은 제안된 NF제어기의 성능의 신뢰성을 확인하기 위하여 속도궤적추종 실험과 마찬가지로 기존의 컴퓨터-토크 제어기법에 대해 10%의 파라미터 불확실성을 고려한 상태에서의 방향각궤적추종 성능실험결과를 나타내고 있다. 컴퓨터-토크 제어기는 10%의 불확실성을 고려한 상태에서 최대 오버슈트량은 약 0.025(rad)이며, 정착시간은 약 1.2초를 나타내고 있다. 시뮬레이션 실험결과에서 보는 바와같이 제안된 NF제어기는

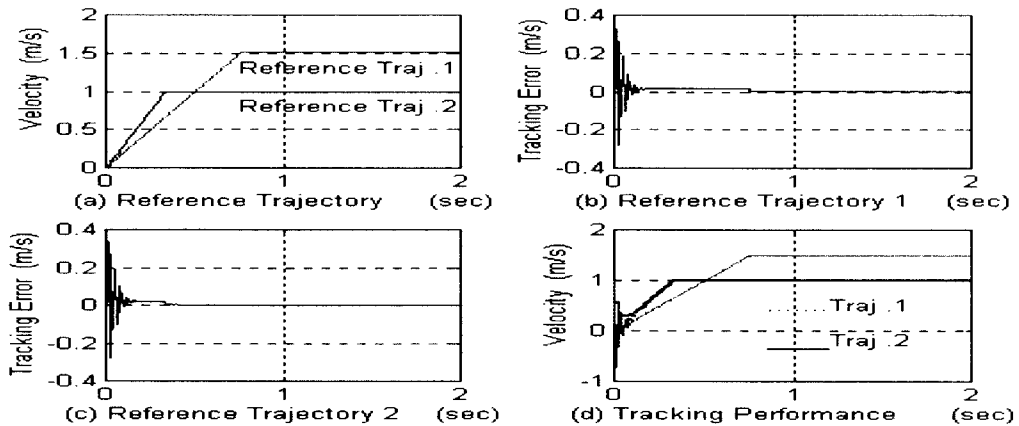


Fig. 6 Control performance of the NF controller for the variation of the velocity trajectory with parameter uncertainties(10%)

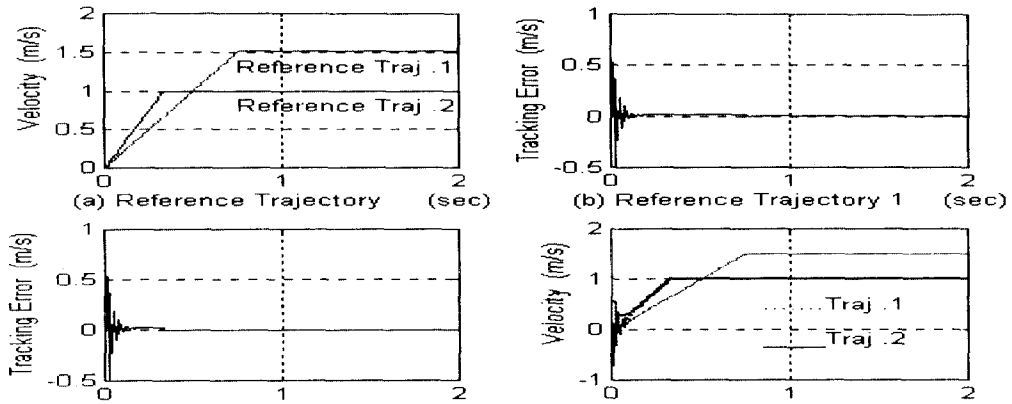


Fig. 7 Control performance of the NF controller for variation of the velocity trajectory with parameter uncertainties (20%)

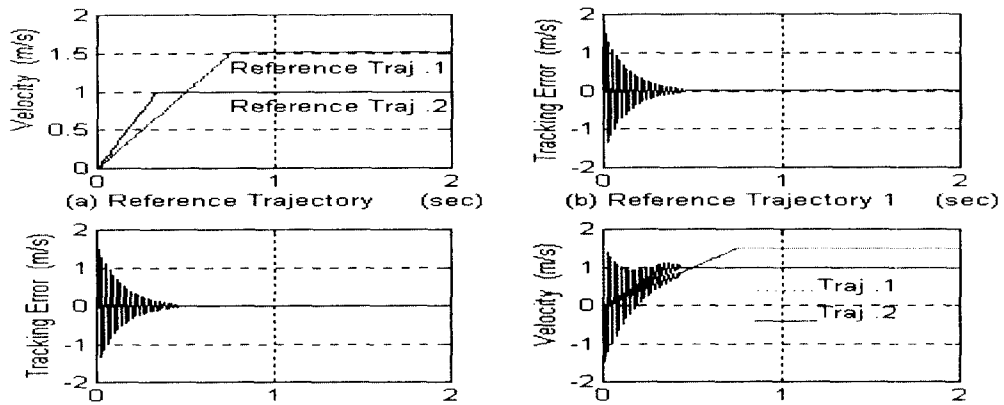


Fig. 8 Control performance of the computer torque method for the variation of the velocity trajectory with parameter uncertainties (10%)

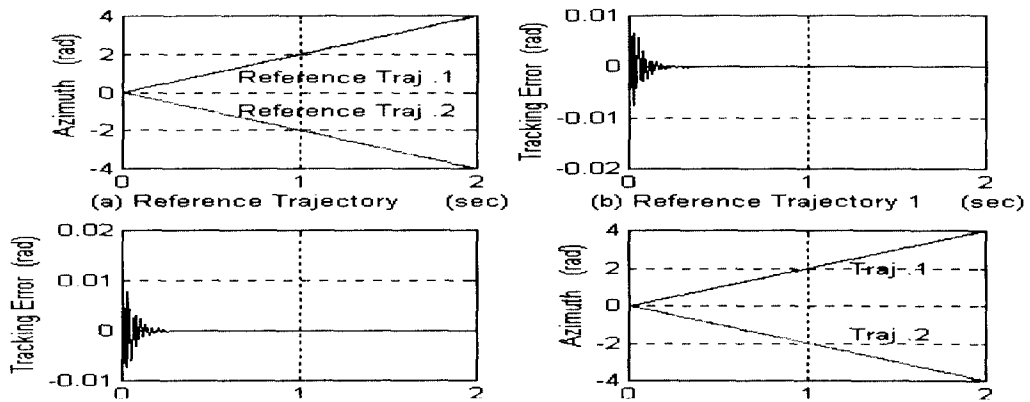


Fig. 9 Control performance of the NF controller for the variation of the azimuth trajectory with parameter uncertainties (10%)

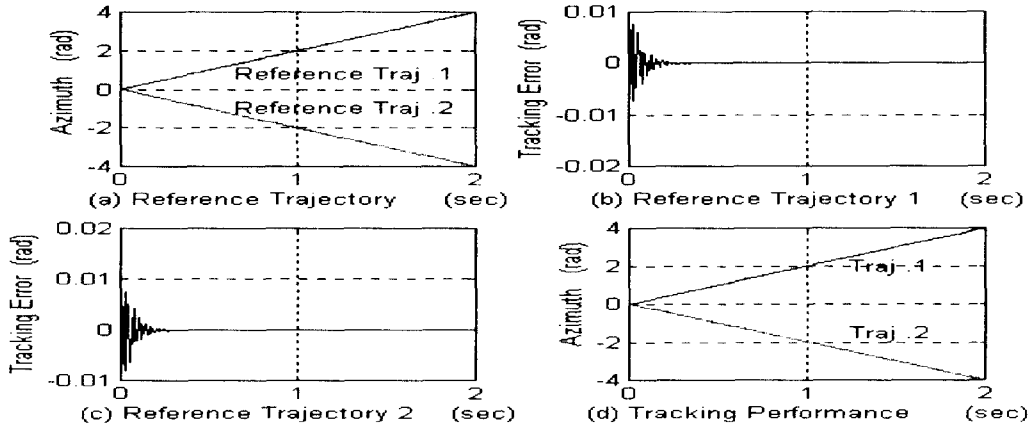


Fig. 10 Control performance of the NF controller for the variation of the azimuth trajectory with parameter uncertainties (20%)

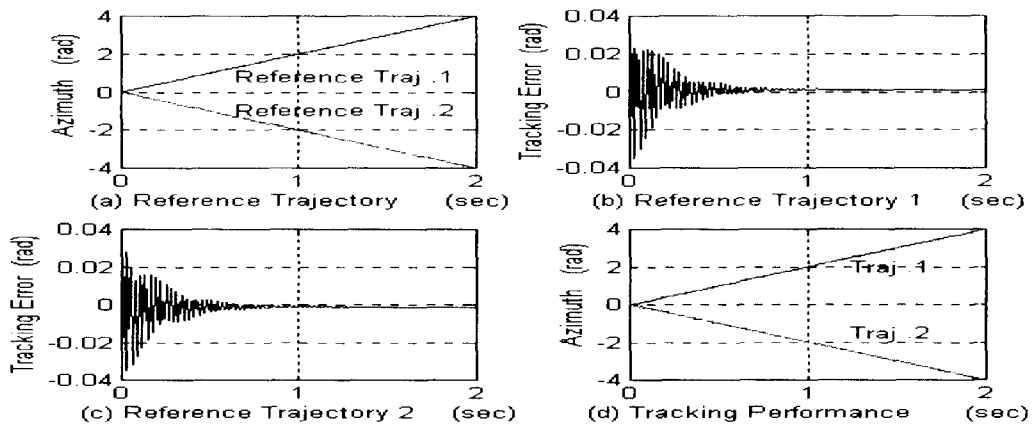


Fig. 11 Control performance of the computer torque method for the variation of the azimuth trajectory with parameter uncertainties (10%)

속도계적추종 및 방향각계적추종에서 컴퓨터 토크 제어기법 보다 파라미터 불확실성에 매우 견실하고 정밀한 제어성능이 유지됨을 확인할 수 있다.

4. 결 론

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 신경제어 구조와 퍼지 추론을 통해 제어입력을 생성하는 퍼지제어 구조를 결합한 새로운 구조의 실시간 제어 가능한 뉴럴-퍼지제어기법을 제안하였다.

제어구조의 특징은 가우스 멤버쉽수를 적용한 퍼지추론법과 역전파 알고리즘을 통한 다층구조의 신경망 제어구조로 설계함으로써 미지 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 이동용 로봇의 속도 및 방향각 제어에 대한 제어 성능 시험을 시뮬레이션을 통하여 예증하였다.

제안된 뉴로-퍼지제어기의 제어성능을 컴퓨터 토오크법의 제어성능과 비교하여 제어성능의 강인성과 정밀성을 검증함으로써 제안된 제어기는 동적특성에 대한 명확한 정보를 알지 못하는 불확실성 시스템 즉, 제어대상인 플랜트의 파라미터변화 및 부하의 변동에 대해서도 능동적이고 우수한 제어 성능이 유지될 수 있는 오프라인 상태에서의 실시간 제어의 가능성을 제시하였다.

제안된 제어구조를 바탕으로 고속처리속도 및 병렬구조를 가지는 디지털 신호처리기인 TMS320C30을 이용한 뉴로-퍼지제어기의 H/W를 구성 제작을 통한 실제 제어시스템을 구축하여 실험을 통한 신뢰성 확인 및 제품화 가능성을 위한 실험의 수행중에 있으며, 또한 해저탐사 및 해저개발을 위한 예비 단계를 위한 실험도 병행할 계획에 있다.

참 고 문 헌

- 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*, Portland, Oregon, July 11-15, 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*, Vol. 1, MIT Press, Mass, 1987
- 5) Horikawa, S., Furuhashi, T., Okuma, 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. et al., "A Fuzzy Controller Using a Neural Network and Its Capability to Learn Expert's Control Rules", *IIZUKA '90*:103-106, 1990
- 7) Psaltis, D., Sideris, A., and Yamamura, A., "A Multilayered Neural Network Controller", *IEEE Control Systems Magazine*, April, pp.17-21, 1988
- 8) Sugeno, M., *Fuzzy Control*, Nikkan-kogyo-shinbun-sha, Tokyo, 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, and The PDP research group (Eds), *Parallel Distributed Processing: Explorations in the Microstructures of Cognition*, Vol. 1: Foundations, MIT Press. Cambridge; MA, 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