

계층적 모듈라 신경망을 이용한 이족로봇 지능제어기

The Intelligent Control System for Biped Robot Using Hierarchical Mixture of Experts

최우경, 하상형, 김성주*, 김용택, 전홍태

Woo-Kyung Choi, Sang-Hyung Ha, Seong-Joo Kim*, Yong-Taek Kim and Hong-Tae Jeon

중앙대학교 전자전기공학부

요약

본 논문에서는 지능제어기법을 이용하여 이족로봇 제어를 설계한다. 이족로봇 제어기는 복잡성을 해결하기 위해 4개 소그룹으로 모듈화 하고, 이 모듈들은 신경망을 이용한 계층적 모듈라 신경망(Hierarchical Mixture of Experts; HME) 기법을 도입한다. 그리고 신경망은 직접제어기법으로 이족로봇의 역 동력학을 학습한다. HME는 나무구조의 네트워크로 입력력 집합을 학습하여 출력공간에 대한 입력공간을 재분할하는 능력을 가지고 있다. EM 알고리즘을 이용한 HME는 반복적 학습을 통하여 이족로봇의 동력학을 모델링하며 HME의 가상오차를 생성하여 이족로봇보행시 안전한 보행을 수행할 수 있는 이족로봇의 제어기를 설계한다.

Abstract

This paper proposes the controller for biped robot using intelligent control algorithm. In order to simplify the complexity of biped robot control, manipulator of biped robot is divided into four modules. These modules are controlled by intelligent algorithm with Hierarchical Mixture of Experts(HME) using neural network. Also neural network having direct control method learns the inverse dynamics of biped robot. The HME, which is a network of tree structure, reallocates the input domain for the output by learning pattern of input and output. In this paper, as a result of learning HME repeatedly with EM algorithm, the controller for biped robot operating safety walking is designed by modelling dynamics of biped robot and generating virtual error of HME.

Key words : Biped Robot, Intelligent Control, Hierarchical Mixture of Experts, Dynamics

1. 서론

최근 각광을 받게 된 로봇은 인간처럼 보행이 가능한 이족로봇이다. 또한, 인간의 존엄성과 가치에 대한 관심이 높아짐에 따라 위험한 장소에서 인간이 수행할 작업을 대신할 수 있는 이족로봇에 대한 요구가 증가하여, 바퀴 구동식 이동(Mobile)로봇과 다리 구동식 보행로봇 등이 연구 되고 있다. 바퀴 구동식 이동로봇은 제작 및 제어가 쉽고 에너지 효율은 높으나 작업장의 바닥면의 굴곡과 요철, 계단 등의 환경에 많은 영향을 받게 되어, 인간중심의 작업환경에서 바퀴 구동식 로봇의 적용은 제한받게 된다. 하지만 인간과 유사한 구조를 갖는 이족로봇은 대부분의 인간생활 환경에서 어려움 없이 작업을 수행할 수 있을 것이다.

1960년대 후반 Vucobratovic이 이족로봇에 관한 모델링을 제안하였다[1]. 이를 기초로 이족로봇에 대한 많은 연구가 수행되었다. 1990년대 후반의 연구들은 이족 걸음새나 로봇의 몸체추진 궤적에 관심을 보이고 있다. 이 연구들은 역 동력학(inverse dynamics) 모델링에 의한 토크 제어를 하거나 신경회로망 기법을 이용한 걸음새를 구현하기도 했다. 이론적 접근방법은 이족로봇의 안정성 및 수학적 모델의 유도가

필수적이거나 해결되지 못하고 있으며 현재 어려운 문제로 남아있다. 그래서 이론적 모델링의 접근방법과는 달리 보행을 위한 구동 메커니즘의 설계와 실험적인 방법으로 접근한 연구들이 있다.

본 논문에서는 동력학의 역 모델링 기법을 이용한 이족로봇의 제어기를 설계하는 것이다. 이족로봇을 4개 관절그룹으로 모듈화 하였다. 그리고 모듈화된 제어대상을 제어하기 위해 통합한 제어기 형태로 계층적 모듈라 신경망(Hierarchical Mixture of Experts; HME)을 사용한다. HME는 학습이 끝난 후 실시간 제어가 가능한 병렬처리 구조이고 계층적 구조로 입력 공간을 여러 개로 군집화하여 기존의 이족로봇의 전문가 지식 바탕의 스위치 역할도 수행 가능하다. 본 연구에서는 전형적인 HME의 학습알고리즘을 수정하여 이족로봇용 제어기 학습알고리즘을 제안하고 확장성이 있고 적응적인 제어기를 설계하고자 한다.

2. 이족로봇 기구학 및 동력학

2.1 일족보행로봇의 기구학 및 동력학

그림 1과 같이 발끝 축의 프레임{A}와 본체 골반의 프레임{B}의 기구학적 연결은 아래와 같다. 관절공간상의 변수 $\theta_1, \theta_a, \theta_k$ 및 θ_h 는 가상 발끝(Toe)관절, 발목(Ankle)관절, 무

접수일자 : 2006년 5월 20일

완료일자 : 2006년 8월 7일

릎(Knee)관절과 골반(Hip)관절의 각도 변수들이며, L_1 과 L_2 는 각 관절의 길이를 나타낸다. L_h 는 뒷굽의 높이를 나타낸다. 기준좌표계(프레임 {O})와 보행로봇 좌표계(프레임 {A})의 관계는 바닥은 평평하다는 가정 하에 ${}^A_0R = I$ 로 정한다. {A}에서 {B}로의 기구학을 정리하면

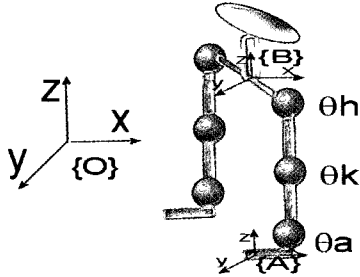


그림 1. 일족보행로봇의 외형도
Fig. 1. Figure of single support leg

$${}^B_A X = \begin{pmatrix} x \\ z \\ \theta \end{pmatrix} = \begin{pmatrix} -L_{vip}c_l - L_h s_l - L_1 s_{l+a} - L_2 s_{l+a+k} \\ -L_{vip}s_l - L_h c_l - L_1 c_{l+a} - L_2 c_{l+a+k} \\ -\theta_h - \theta_k - \theta_a - \theta_l \end{pmatrix} \quad (1)$$

여기서 $c_a \equiv \cos(\theta_a)$, $s_a \equiv \sin(\theta_a)$ 이다. 이를 미분하여 자코비안 행렬을 만들면 다음과 같다.

$${}^B_A J = \begin{pmatrix} J_{1,1} & J_{1,2} & J_{1,3} & 0 \\ J_{2,1} & J_{2,2} & J_{2,3} & 0 \\ -1 & -1 & -1 & -1 \end{pmatrix} \quad (2)$$

자코비안 행렬은 프레임{A}와 프레임{B} 사이의 관절속도와 직교좌표계 속도의 관계를 가지고 있다. ${}^B_A X = {}^B_A J \theta$ 이며 관절 토크의 힘의 관계는 아래와 같다.

$$\tau = ({}^B_A J)^T ({}^B_A F), (\tau: \text{관절토크 벡터}, F: \text{힘 벡터}) \quad (3)$$

여기서 τ 는 관절토크 벡터이며, F 는 힘 벡터이다. 발끝 관절에는 모터가 존재치 않기 때문에 $\tau_l = 0$ 이다. 이런 조건을 식 (1)에 적용하면 아래와 같아진다.

$$\begin{pmatrix} 0 \\ \tau_a \\ \tau_k \\ \tau_h \end{pmatrix} = \begin{pmatrix} J_{1,1} & J_{1,2} & J_{1,3} & -1 \\ J_{2,1} & J_{2,2} & J_{2,3} & -1 \\ J_{1,3} & J_{2,3} & -1 & -1 \\ 0 & 0 & -1 & -1 \end{pmatrix} \begin{pmatrix} f_x \\ f_z \\ f_\theta \end{pmatrix} \quad (4)$$

f_z 와 f_θ 를 열거하여 f_x 로 식을 전개하면 다음과 같다.

$$f_x = \frac{-J_{2,1}}{J_{1,1}} \frac{1}{J_{1,1}} \begin{pmatrix} f_z \\ f_\theta \end{pmatrix} \quad (5)$$

$$\begin{pmatrix} \tau_a \\ \tau_k \\ \tau_h \end{pmatrix} = \begin{pmatrix} \frac{-J_{1,2}J_{2,1}}{J_{1,1}} + J_{2,2} \frac{J_{1,2}}{J_{1,1}} - 1 \\ \frac{-J_{1,3}J_{2,1}}{J_{1,1}} + J_{2,2} \frac{J_{1,3}}{J_{1,1}} - 1 \\ 0 \end{pmatrix} \begin{pmatrix} f_z \\ f_\theta \end{pmatrix} \quad (6)$$

여기서 모든 관절각도는 계측 가능하다고 가정하며, 발끝에 액추에이터가 없으므로 “가상의” 라는 단어를 사용하였으며, 발끝의 각도는 아래의 식으로 계산된다.

$$\theta_l = -\theta - \theta_h - \theta_k - \theta_a \quad (7)$$

2.2 이족로봇의 완성

기존의 J. Pratt의 논문[2]에서 제한행렬(constraint matrix)를 구축하고 임의의 힘벡터 F_x, F_z 를 가지고 이족간의 힘분배 문제를 해결하였다.

$$\begin{pmatrix} f_z \\ f_\theta \end{pmatrix}_{left} = \alpha \begin{pmatrix} f_z \\ f_\theta \end{pmatrix}_{right}, \quad \begin{pmatrix} f_z \\ f_\theta \end{pmatrix}_{right} = (1 - \alpha) \begin{pmatrix} f_z \\ f_\theta \end{pmatrix} \quad (8)$$

여기서 α 는 $0 < \alpha < 1$ 의 범위에 있다. 일족의 경우처럼 x 방향의 힘을 줄 수 없지만 z 와 θ 의 방향으로 힘벡터를 조정할 수 있으며 이족간의 힘 분배를 결정하고 식 (8)을 이용하여 각 관절의 토크를 계산할 수 있다.

3. 신경망을 이용한 로봇 제어기

3.1 계층적 모듈라 신경망의 구조

HME는 Jacobs와 Jordan[3][4][5]이 제안한 모듈라 신경망(Modular Neural Network; MNN)의 구조를 이용 하였다. MNN은 엑스퍼트 네트워크(Expert Network; EN)이라고 불리는 M개의 교사모듈(Supervised Module)과 각각의 EN 사이에 중개역할을 담당하는 게이팅 네트워크(Gating Network; GN)로 구성되어 있다.

HME 구조는 그림 2와 같이 트리구조의 GN와 EN으로 구성된다. HME은 트리의 나뭇잎 위치에 EN을 배치하여 지역적 문제의 해를 푸는 역할을 수행한다. EN의 출력들이 전체 문제의 해를 구성할 수 있도록 GN에 의해 계층적으로 결합된다.

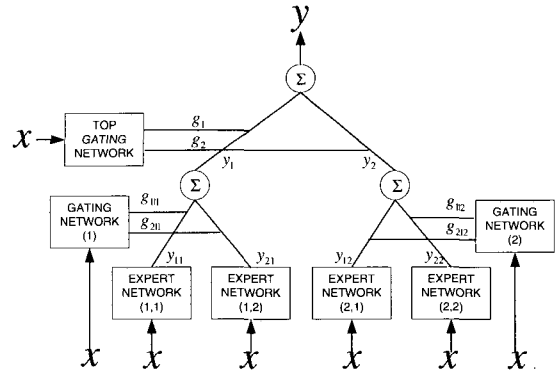


그림 2. 계층적 모듈라 신경망(HME)
Fig. 2. Hierarchical Mixture of Expert Network

입출력 학습데이터($T = (x_i, y_i), i = 0, 1, 2, \dots, N$)을 기초로 입출력 매핑을 수행한다. GN와 EN은 같은 x_i 를 입력으로 사용하나, EN은 원하는 출력을 이끌어 내도록 신뢰할 만한 출력 값을 계산하고, GN은 EN의 출력 값에 신뢰할 만한 가중치 벡터 값들을 제공한다. 계층적 모듈라 신경망(HME)은 GN을 이용하여 확률적으로 선택함으로써 데이터를 분류한다.

N 집단에 속해 있는 입력 벡터 $x^{(t)}$ 을 관찰할 때 원하는

출력벡터 $y^{(t)}$ 을 구축하면 다음 식을 얻는다.

$$y = \sum_{i=1}^N g_i \sum_{j=1}^M g_{ji} \mu_{ji} \quad (9)$$

여기서 μ_{ji} 는 하위 단계의 i 번째 모듈의 j 번째 EN의 출력이며, g_{ji} 와 g_i 는 GN의 출력이다. M 은 하위단계 수준 모듈에서 어느 클러스터 안의 EN의 개수이고 N 은 상위 수준 모듈의 노드의 개수이며 클러스터의 개수이다. EN은 MLNN형태의 신경망을 사용한다. EN의 입력 x 에 대하여 EN의 출력을 다음과 같다.

$$\mu_{ji} = f(U_{ji}x) \quad (10)$$

여기서 U_{ji} 는 EN의 가중치행렬, $f(\cdot)$ 는 활성화함수이다. GN의 출력은 아래와 같이 얻을 수 있다.

$$\xi_{ij} = v_{ij}^T x, \quad g_{ji} = \frac{\exp(\xi_{ij})}{\sum \exp(\xi_{ij})} \quad (11)$$

여기서 v_{ij}^T 는 GN의 가중치 벡터이며, 학습이 끝난 후 g_i 와 g_{ji} 는 후건부 확률 $P(i|x)$ 와 $P(j|i, x)$ 을 제공한다. 식 9에 식 10을 적용하면 다음의 식으로 정리된다.

$$y = \left(\sum_{i=1}^N g_i \sum_{j=1}^M g_{ji} U_{ji} \right) x \quad (12)$$

만약 g_i 와 g_{ji} 가 상수이면, y (식 12)는 HME의 자코비안 행렬로 볼 수 있다. HME가 시스템의 전방향 모델 역할을 한다. 역 모델링 학습이 가능하고 학습이 되었다면 원하는 출력신호(y^d)을 얻을 수 있다. 그러므로 y (식 12)을 얻으면, 신경망의 정확한 오차 값을 얻을 수 있다.

$$y^d = \left(\sum_{i=1}^N g_i^d \sum_{j=1}^M g_{ji}^d U_{ji} \right) x^d \quad (13)$$

상위 GN의 i 번째 출력 g_i 는 다음과 같이 정의된다.

$$g_i = \frac{\exp(u_i)}{\sum_{j=1}^K \exp(u_j)}, \quad i = 1, 2, \dots, K \quad (14)$$

여기서 u_i 는 입력벡터 x 에 대한 가중치 합이다. 하위 GN의 i 번째 출력함수는 Soft-Max 함수로 다음과 같다.

$$g_{ji} = \frac{\exp(u_{ji})}{\sum_{k=1}^L \exp(u_{ki})}, \quad \begin{cases} i = 1, 2, \dots, L \\ j = 1, 2, \dots, K \end{cases} \quad (15)$$

u_{ji} 는 i 번째 클러스터의 특정 뉴런에 적용된 입력의 가중치 합이다. 각 클러스터의 EN들은 상기에서 언급한 모듈라 신경망의 EN을 사용한다.

$$y_i = \sum_{j=1}^L g_{ji} y_{ij}, \quad i = 1, 2, \dots, K \quad (16)$$

HME의 출력벡터는 아래의 식과 같이 정의된다.

$$y = \sum_{i=1}^K g_i y_i \quad (17)$$

3.2 확률해석

학습데이터 $\{x, y\}$ 는 Jordan & Jacobian[6]에서 제안한 nested regressive process에 의해 만든다. 기준응답 d 는 다음의 회귀응답으로부터 발생된다.

$$d = F_{\bar{j}}(x) + \epsilon, \quad \begin{cases} i = 1, 2, \dots, K \\ j = 1, 2, \dots, L \end{cases} \quad (18)$$

여기서 $F_{\bar{j}}$ 는 임의의 표본함수의 과거의 값을 이용하여 미래의 값을 정확하게 예측할 수 있는 결정적인 함수이다. ϵ 는 영점 평균인 가우시안 분포 확률벡터이다. 학습데이터 $\{x, d\}$ 에 대한 확률분포를 모델링하는 것이 목적이다. 이 목적함수가 직접적인 학습문제로 근사되기 위해 동반 가우시안 혼합 모델의 확장된 형태로써 정의된 로그우도함수를 이용하며 아래의 식과 같다.

$$l = \ln \sum_{i=1}^K g_i \sum_{j=1}^L g_{ji} \exp\left(-\frac{1}{2} d - y_{ji}^2\right) \quad (19)$$

여기서 d 는 입력벡터 x 에 대한 기준응답벡터이다. 회귀의 경우, 확률모델은 가우시안분포라는 가정을 한다. 각 EN을 위한 $\sigma^2 I$ 의 형태의 식별 공분산 행렬은 다음의 계층적 확률모델을 만든다.

$$P(y|x, \theta) = \frac{1}{(2\pi)^{n/2} \sigma^n} \sum_i g_i \sum_j g_{ji} e^{-\frac{1}{2\sigma^2} (y-y_{ji})^T (y-y_{ji})} \quad (20)$$

g_i, g_{ji} 및 y_{ji} 는 미지의 값들이고 미지의 조건부 확률이며 모든 확률변량은 입력벡터 x 에 대하여 조건부이다. 이런 조건에 대한 학습 알고리즘을 만들기 위해 3가지 사후확률을 정의한다.

첫 번째는 i 번째 클러스터에서 기준응답벡터 d 을 만드는 조건부 사후확률을 정의하며 아래의 식과 같다.

$$h_i = \frac{g_i \sum_{j=1}^L g_{ji} \exp\left(-\frac{1}{2} d - y_{ji}^2\right)}{\sum_{i=1}^K g_i \sum_{j=1}^L g_{ji} \exp\left(-\frac{1}{2} d - y_{ji}^2\right)}, \quad i = 1, 2, \dots, K \quad (21)$$

두 번째는 i 번째 클러스터의 j 번째 EN에서 기준응답벡터 d 를 만드는 조건부 사후확률을 정의하며 아래의 식과 같다.

$$h_{ji} = \frac{g_{ji} \exp\left(-\frac{1}{2} d - y_{ji}^2\right)}{\sum_{j=1}^L g_{ji} \exp\left(-\frac{1}{2} d - y_{ji}^2\right)}, \quad \begin{cases} i = 1, 2, \dots, K \\ j = 1, 2, \dots, L \end{cases} \quad (22)$$

세 번째로 h_i 와 h_{ji} 사이의 곱을 접합 사후확률(Joint a Posterior Probability)로 정의하며 아래의 식과 같다.

$$h_{\bar{j}} = h_i \times h_{ji} = \frac{g_i g_{ji} \exp\left(-\frac{1}{2} d - y_{ji}^2\right)}{\sum_{i=1}^K g_i \sum_{j=1}^L g_{ji} \exp\left(-\frac{1}{2} d - y_{ji}^2\right)} \quad (23)$$

집합 사후확률 h_{ji} 는 아래의 조건을 만족하여야 한다.

$$\sum_{j=1}^L \sum_{i=1}^K h_{ji} = 1, 0 \leq h_{ji} \leq 1 \text{ for all } (i, j) \quad (24)$$

사후확률 h_i, h_{ji} 는 입력벡터 x 와 기준응답벡터 d 에 대하여 조건부 확률이다. 출력벡터 y_{ji} 와 g_i, g_{ji} 는 EN과 GN을 구성하고 있는 신경망의 연결 가중치에 의존한다. 로그우도 함수 l 은 미지의 파라미터로써 신경망 시냅스 가중치로 표현 가능한 함수로 볼 수 있다. 이런 이유로 로그우도함수를 최대화 한다는 것은 파라미터 최대우도추종을 이끌어낸다. 시스템에서 모든 시냅스 가중치에 즉각적으로 작용하는 적은양의 계산량을 위해 기울기상승기법을 이용하여 l 함수의 최대화는 수행할 수 있다. Jordan & Jacob의 방식[7]을 이용하여 시냅스의 조정을 계산한다.

3.3 HME 확률적 기울기 학습 알고리즘

학습알고리즘은 다음과 같은 $\frac{\partial l}{\partial a_i}, \frac{\partial l}{\partial c_{ji}}$ 및 $\frac{\partial l}{\partial w_{ji}^{(m)}}$ 등의 요소들을 결정하여야 한다. 여기서 a_i 는 상위 GN의 i 번째 출력 신경망의 가중치 벡터이고, c_{ji} 는 i 번째 클러스터 GN의 j 번째 출력 뉴런의 시냅스 가중치 벡터로 정의 한다. $w_{ji}^{(m)}$ 은 i 번째 클러스터의 j 번째 EN의 q 개의 출력 뉴런 중 m 번째 출력 뉴런 가중치 벡터로 정의한다.

$$\frac{\partial l}{\partial a_i} = \frac{\partial l}{\partial u_i} \frac{\partial u_i}{\partial a_i}, i = 1, 2, \dots, K \quad (25)$$

위의 식을 유도하면 아래의 식으로 정리된다.

$$\Delta a_i = \eta \sum_t (h_i^{(t)} - g_i^{(t)}) x^{(t)} \quad (26)$$

로그우도함수를 i 번째 클러스터 GN의 j 번째 출력 뉴런의 시냅스 가중치 벡터로 편미분하면

$$\frac{\partial l}{\partial c_{ji}} = \frac{\partial l}{\partial u_{ji}} \frac{\partial u_{ji}}{\partial c_{ji}}, \begin{cases} i = 1, 2, \dots, K \\ j = 1, 2, \dots, L \end{cases} \quad (27)$$

위의 식을 유도하면 아래의 식으로 정리된다.

$$\Delta c_{ji} = \eta \sum_t h_i^{(t)} (h_{ji}^{(t)} - g_{ji}^{(t)}) x^{(t)} \quad (28)$$

로그우도함수를 i 번째 클러스터의 j 번째 EN의 q 개의 출력 뉴런 중 m 번째 출력 뉴런 가중치 벡터로 편미분하면,

$$\frac{\partial l}{\partial w_{ji}^{(m)}} = \frac{\partial l}{\partial y_{ji}^{(m)}} \frac{\partial y_{ji}^{(m)}}{\partial w_{ji}^{(m)}}, \begin{cases} i = 1, 2, \dots, K \\ j = 1, 2, \dots, L \\ m = 1, 2, \dots, q \end{cases} \quad (29)$$

위의 식을 유도하고 정리하면 아래의 식과 같다.

$$\Delta w_{ji}^{(m)} = \eta \sum_t h_i^{(t)} h_{ji}^{(t)} (y_{ji}^{(t)} - \mu^{(t)}) x^{(t)T} \quad (30)$$

여기서 η 는 학습율이고, 상기 식을 통하여 학습에 필요한 요소들을 결정할 수 있다.

4. 이족로봇의 제어기

제안된 이족로봇제어기는 HME 역 모델링 제어기로 구성되어 있다. HME의 입력은 궤적생성기로부터 나오는 관절 각도이고 출력은 이족로봇의 관절토크이다. HME의 사후확률은 시스템의 오차신호를 자코비안에 의해 역 모델링 된 신호로부터 구해진다. HME은 EM알고리즘을 사용하여 사후확률과 사전확률의 차이가 감소하도록 한다. 즉 시스템의 오차가 줄어드는 방향으로 학습한다.

여러 가지의 신경망 역 모델링 기법들 중 적용된 부분을 수식으로 정리하면 아래와 같다.

$$y = N(Y_d, W) \quad (31)$$

이족로봇의 시스템 함수는 아래와 같이 정의한다.

$$y_s = S(y) \quad (32)$$

HME은 출력을 다시 입력으로 사용하여 역 모델링을 수행한다. 역 모델링을 수행할 때 오차($e = y^s - y^d$)를 가지며 오차를 시스템의 자코비안 관계로 표현하면 아래 식과 같이 실행지수(E)로 표현된다.

$$E = y^s - y^d \quad (33)$$

위의 실행지수 E 을 기울기 감소 가중치 갱신 규칙을 사용하면 다음 식과 같이 유도된다.

$$\Delta w = -\eta \left(\frac{\partial N}{\partial w} \right) \Delta y \quad (34)$$

Δy 는 아래와 같이 쓸 수 있다.

$$\Delta y = \frac{d(S^T)}{dt} (y^s - y^d) = J(y)^T (y^s - y^d) \quad (35)$$

앞에서 언급한 기준출력(y^d)을 구하면 신경망의 정확한 오차를 구할 수 있다. 이를 위해, 새로운 GN으로 Y-GN을 그림 3과 같이 추가한다.

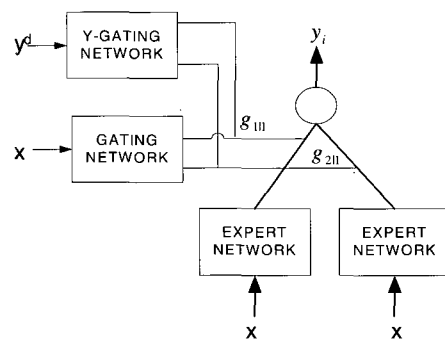


그림 3. Y-GN을 포함한 모듈라 구조

Fig. 3: The Structure of Modular NN with Y-GN

추가된 GN은 입력으로 기준 출력을 받기 때문에 Y-GN이라 칭한다. 전방향 모델링과정이나 역방향 학습과정은 일반 GN으로 학습을 수행하고, 학습이 끝난 후 Y-GN의 출력 g_i^y 와 g_{ji}^y 는 사후확률 $P(i|y^d)$ 와 $P(j|i, y^d)$ 을 각각 제공한다. 기준출력(y^d)을 얻는 입력(x)이 일반해라면, $g^y = g_i$

의 관계가 성립한다. 그러나 자코비안 행렬을 처음부터 끝까지 g_i^y 와 $g_{j_i}^y$ 을 이용하여 구할 수 없기 때문에 경우에 따라서 g_i^y 는 항상 g_i 와 같지는 않다. Y-GN의 출력($g_{j_i}^y$)이 영(0)이 아니면, μ_{j_i} 는 자코비안 행렬의 요소가 된다.

이런 요소에 의해 EN의 출력값(y_i)과 기준출력(y^d)의 오차를 이용하여 역 모델링 학습을 위한 가중치 갱신 규칙을 사용하면, 아래 식과 같은 결과를 얻을 수 있다.

$$\Delta y = \sum_{i=1}^N \frac{g_i^y \exp(-y^d - y_k^2/2)}{\sum_{k=1}^N g_k^y \exp(-y^d - y_k^2/2)} \times \sum_{j=1}^M \frac{g_{j_i}^y \exp(-y^d - y_{j_i}^2/2)}{\sum_{l=1}^M g_{l_i}^y \exp(-y^d - y_{l_i}^2/2)} U_{j_i}^T (y^s - y^d) \quad (36)$$

여기서 얻은 Δy 를 식 34에 대입하면 EN의 기울기 갱신 값을 얻을 수 있다.

5. 모의실험

모의실험은 모의실험 I과 모의실험 II로 나눈다. 모의실험 I은 신경망의 성능비교 결과이고 모의실험 II는 7관절로 구성된 평면 이족로봇의 모델을 실험한 결과를 보여준다.

5.1 모의실험 I

신경망들이 학습할 모델함수는 아래 식들과 같다. 신경망 모의실험에서 다중 입력 다중 출력(MIMO) 형태의 모델함수를 채택한다. 모델함수는 아래의 식과 같다.

$$Y = F(X) \quad (37)$$

입력신호($X = [x_1, x_2]$)는 원 좌표의 집합으로 구성한다.

$$x_1 = \sin(t), \quad x_2 = \cos(t) \quad (38)$$

기준 출력($Y = [y_1, y_2, y_3, y_4, y_5]$)은 아래와 같이 다섯 개의 신호로 구성하였다.

$$\begin{aligned} y_1 &= \sin(x_1) \cos\left(\frac{x_1}{3}\right) \cos\left(\frac{1}{20} \cos\left(\frac{x_1}{7}\right)\right), \\ y_2 &= \cos(x_1) \cos\left(\frac{x_1}{2}\right) \cos\left(\frac{1}{3} \cos\left(\frac{x_1}{5}\right)\right), \\ y_3 &= \cos(x_1) \sin\left(\frac{x_1}{5}\right) \cos\left(\frac{1}{3} \cos\left(\frac{x_1}{15}\right)\right), \\ y_4 &= \cos(x_2) \sin\left(\frac{x_1}{20}\right) \cos\left(\frac{1}{3} \cos\left(\frac{x_1}{25}\right)\right), \\ y_5 &= \cos(x_2) \cos\left(\frac{x_1}{20}\right) \cos\left(\frac{1}{3} \cos\left(\frac{x_1}{50}\right)\right) \end{aligned} \quad (39)$$

모의실험 I에서 사용한 신경망의 종류는 다음 표와 같다.

표 1. 신경망의 종류
Table 1. Type of Neural Network

Network	다층신경망	MNN	HME
입력노드	2	2	2
은닉층 노드	120	30	30
출력층 노드	5	5	5
EN개수	-	4	4
GN개수	-	1	상위 1, 하위 1
학습방법	BP 알고리즘	EM 알고리즘	EM 알고리즘

상기의 입출력 관계를 다층 신경망, 모듈라 신경망 및 HME로 나누어 학습한 결과는 그림 4에서 보여준다.

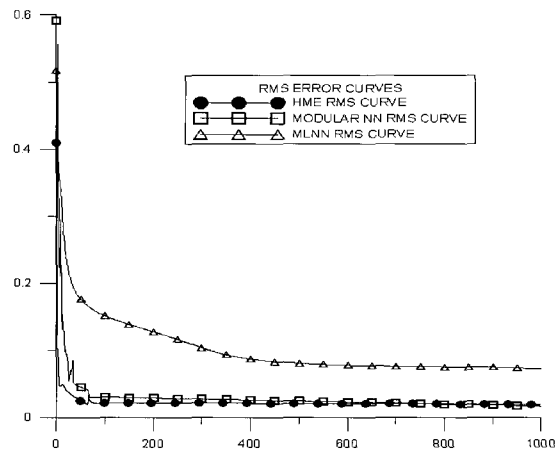


그림 4. 신경망 오차 비교곡선
Fig. 4. The RMS Error Curves of MLNN, MNN and HME

다층신경망은 3000번 반복 학습을 수행 시에도 RMS 오차가 0.07이하로 내려가지 않는다. 즉 다층 신경망은 전역적 안정성을 보장하지 못하며 입력의 변화에 민감하게 반응하지 못한다. 이에 비해 모듈라 신경망과 HME는 입력패턴에 따라 적절하게 반응하는 EN들을 EM 알고리즘으로 선택하고 있기 때문에 입력공간을 분할하여 각 EN들이 모델링을 수행한다. 특히, HME는 신경망에 비해 학습가능이나 수렴속도가 매우 좋은 장점을 가지고 있으며 모듈라 신경망에 비해 좀더 정확한 입력공간 분할 능력을 가지고 있다. 또한 입출력의 관계를 나누어 학습을 수행하기 때문에 왜란에 강한 강점을 가진다.

다음의 모의실험 II는 HME를 이용한 실험결과이다. 입력(X)의 궤적은 그림 5와 같다.

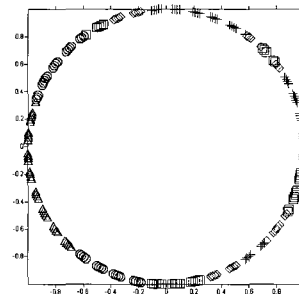


그림 5. 입력 x_1, x_2 의 궤적
Fig. 5. The Trajectory of x_1, x_2

그림 6은 HME의 상위 GN의 사전확률 분포와 하위 사전 확률 분포 사이의 관계를 보여준다. 상위 GN에서 입력공간을 분할하고 하위 GN에서 조밀한 입력공간을 분할하고 있음을 알 수 있다. 또한 그림 7을 보면 학습이 끝난 후 사전확률과 사후확률이 같아지는 것을 알 수 있다.

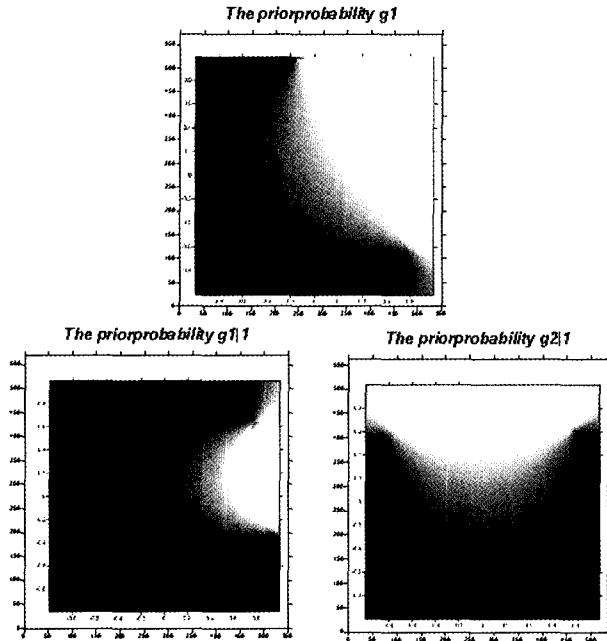


그림 6. HME 사전확률분포 1

Fig. 6. HME image map of prior probability distribution 1

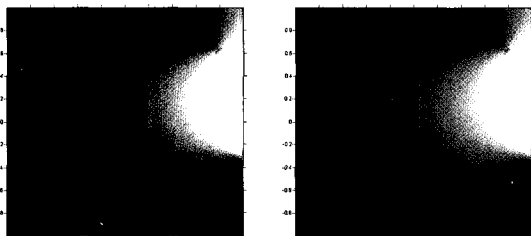


그림 7. 사전확률 g_{11} 와 사후확률 h_{11} 의 분포

Fig. 7. Distribution of prior prob. g_{11} and posterior prob. h_{11}

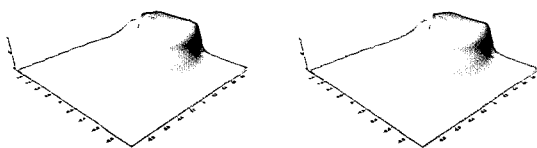


그림 8. 사전확률 g_{11} 와 사후확률 h_{11} 의 3D

Fig. 8. 3D surface of prior prob. g_{11} and posterior prob. h_{11}

5.2. 모의실험 II (로봇의 제어)

모의실험으로 사용할 평면 이족로봇의 사양은 표 2와 같다.

표 2. 이족로봇의 사양
Table 2. Specification of biped robot

파라미터	값	파라미터	값
전체질량	23 Kg	몸체질량	3 Kg
몸체길이	1.2 m	허벅지무게	5 Kg
장단지 무게	4 Kg	발바닥 무게	1 Kg
허벅지 길이	0.5 m	장단지 길이	0.5 m
허벅지 탄성모멘트	0.013 Kg m ²	몸체 탄성모멘트	0.2 c
장단지 탄성모멘트	0.0095 Kg m ²	발바닥 탄성모멘트	0.0014 Kg m

시뮬레이터는 OpenGL을 이용하여 개발하였다. 이족로봇의 모양과 이족로봇 보행상태 전환에 관해 그림 9에서 보여준다.

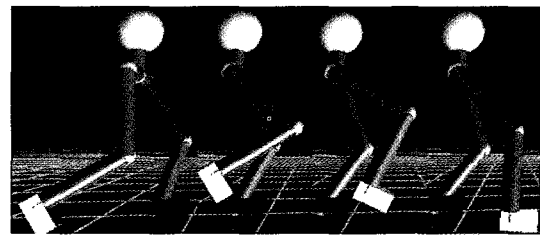


그림 9. 이족로봇 보행상태 전환

Fig. 9. The State Switching of Planner Biped Robot Locomotion

이족로봇의 보행실험 결과는 아래의 그림 10와 같다. 그림 10은 HME 제어기로 구동된 이족로봇의 오차 곡선과 이동로봇의 토크 곡선을 나타낸다.

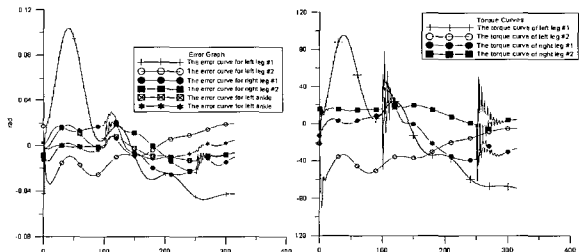


그림 10. 이족로봇의 오차곡선과 토크곡선

Fig. 10. The Error Curves and the Torque Curves of Joint Angle of Biped Robot

6. 결론 및 향후 연구 방향

본 논문에서는 이족로봇의 신경망 제어기 설계를 제안하였다. 기존의 이족로봇 제어는 다중 제어방식으로 다수의 제어기로 구성되어 있다. 보행을 위해선 전문가 지식에 의존하며 고전적인 제어기를 채택했을 때는 제어기 마다 동특성에 맞는 파라미터를 각각 설정하여야 하는 문제가 있다. 파라미터 설정도 쉽지 않은 일이며 적응적이고, 강건한 제어기를 기대하려면 제어기의 구조가 매우 복잡해진다. 이런 문제들을 해결하기위해 신경망을 도입하였고 로봇의 동역학을 학습하여 보행로봇의 제어기 파라미터 설정의 문제를 해결하고자

저 자 소 개

하였다. 또한 HME 신경망의 빠른 수렴속도를 이용하여 보행의 안정성을 보장하고자 하였다. 복잡한 제어대상을 여러 개의 간단한 구조의 제어대상으로 만들어 스위치 구조의 기존의 제어방식 대신 하나의 HME로 이족로봇 제어기를 구성하였다. 본 논문에서 제안한 제어기는 입력 패턴과 출력 패턴에 의존하여 시스템의 역 모델링을 수행한 신경망 제어기 설계를 목적으로 한다. 이를 위해 계층적 모듈화된 HME 신경망을 채택하여 이족로봇의 기존 제어기의 파라미터 설정 및 보행의 안정성 등의 여러 가지 난점을 어느 정도 해소하였다. 상태전환 관련 전문가 지식베이스 기반 스위치 기능을 보행 패턴에 맞춰 HME에서 적절한 제어 값을 만들며 모의 실험을 통하여 이를 입증하였다.

로봇 제어기에서는 안정성 문제가 중요한 문제이므로 향후 이 문제를 연구할 필요가 있다. 또한 파라미터 설정 및 복잡한 전문가 판단기준의 문제를 쉽게 해결 할 수 있는 이족 보행로봇의 제어기를 퍼지 규칙베이스로 구축하는 작업을 향후 연구 분야로 정할 필요가 있다.

참 고 문 헌

[1] M Vukobratovic and A. A. Frank, "Legged locomotion studies," Advances in External Control of Human Etremities, M. H. Gavrilovic and A. B. Wilson Jr., eds. Belgrade: Yugoslav Committee for Electronics and Automation, pp. 407-430, 1969.

[2] Pratt, J., Torres, A., Dilworth, P. & Pratt, G., "Virtual Actuator Control," IEEE International Conference on Intelligent Robots and Systems, 1996.

[3] Simon Haykin, Neural Networks - A Comprehensive Foundation, Macmillian College Publishing Company Inc., 1994.

[4] R. A. Jacobs, M. I. Jordan, s. J. Nowlan, and G. E. Hinton, "Adaptive Mixtures of Local Experts," Neural Computation, Vol. 6., pp. 181-214, 1994.

[5] Craig, J. J., "Introduction to Robotics: Mechanics and Control," addison-Wesley, 1988.

[6] Michael I. Jordan., and R.A. Jacobs, "Hierarchies of Adaptive experts," In Advanced in Neural Information Processing System 4 (J.E. Moody, S.J. Hanson and R.P. Lippmann, eds) San Mateo, CA: Morgan Kaufmann, pp.985-992. 1992.

[7] Michael I. Jordan and Robert A. Jacobs, "Hierarchical Mixtures of Experts and the EM Algorithm," Neural Computation, Vol. 6, No. 1, pp. 181-214, 1994.



최우경(Woo-Kyung Choi)

2000년 : 전남대학교 공학사
 2004년 : 중앙대학교 전자전기공학부
 공학석사
 2004년 ~ 현재 : 동 대학원 박사과정

관심분야 : 소프트 컴퓨팅, 웨어러블 컴퓨팅, 모바일 로봇, 지능시스템

Phone : 02-820-5297
 Fax : 02-817-5508
 E-mail : chwk001@wm.cau.ac.kr

하상형(Sang-Hyung Ha)

1990년 : 중앙대학교 전자공학과 공학사
 1994년 : 동 대학원 전자공학과 공학석사
 2002년 ~ 현재 : 동 대학원 박사과정
 현재 : 휴노랩스 재직중

관심분야 : 소프트 컴퓨팅, 웨이블릿, 인공생명

김성주(Seong-Joo Kim)

제 15권 제 1호 참조

김용택(Yong-Taek Kim)

제 12권 제 6호 참조

전홍태(Hong-Tae Jeon)

제 15권 제 1호 참조