

뉴로-퍼지 제어를 이용한 도립역진자의 각도 및 위치제어

이 근형*, 정 슬**

* 충남대학교 메카트로닉스공학과 (TEL : 042-821-7232, E-mail : sadthink@paran.com)

** 충남대학교 메카트로닉스공학과 (TEL : 042-821-7232, E-mail : jungs@cnu.ac.kr)

Control of an angle and a position of inverted pendulum system using a neuro-fuzzy controller

Geun-Hyeong Lee* and Seul Jung**

*Department of Mechatronics Engineering, BK21 Mechatronics Group, Intelligent Systems & Emotional Engineering Laboratory, Chungnam National University, Daejeon, Korea (Tel : +82-42-821-7232 ; E-mail : sadthink@paran.com)

**Department of Mechatronics Engineering, BK21 Mechatronics Group, Intelligent Systems & Emotional Engineering Laboratory, Chungnam National University, Daejeon, Korea(Tel : +82-42-821-7232 : E-mail : jungs@cnu.ac.kr)

Abstract - 본 논문에서는 도립 역진자 시스템에서의 진자의 도립 상태를 유지하도록 하기 위하여, DSP와 FPGA를 결합하여 ANFIS 뉴로퍼지 제어기를 구현하여 실험하였다. 도립진자의 위치 추종 성능을 PID 제어기와 비교 평가하였다.

1. 서 론

최근의 IT기술의 발달은 시스템이 점차적으로 지능화되어 가는데 촉매로 작용하고 있다. 산업공정에서나 로봇 시스템에서 하드웨어부분은 많은 진전이 있으나 소프트웨어 기술은 그에 미치지 못하고 있는 실정이다. 특히, 차세대 소프트웨어의 꽃이라 할 수 있는 지능기술은 현재 관심이 높아지고 있으며 많은 투자가 예상된다.

지능기술의 발전은 시스템과 접목하여 지능시스템이란 분야를 개척하였으며, 앞으로 좋은 성과가 기대된다. 특히, 지능제어 시스템분야에서의 연구는 지능 알고리즘을 시스템에 구현하므로 실제적인 문제점들을 해결하고자 한다. 대표적인 지능 기술로는 신경회로망, 퍼지 논리, 그리고 진화연산 기술 등이 있다[1,2]. 하지만, 신경회로망은 분석의 어려움이 있고, 퍼지 논리는 퍼지 법칙을 설정하는 어려움이 있고, 진화연산은 실시간에 적용하는 단점이 있다. 따라서, 최근에는 이러한 지능 기술들은 접목하여 단점을 보완하고 장점을 극대화하여 성능을 높이고자 하는 연구가 활발하다. 그 중 하나가 뉴로-퍼지 제어기술이다.

뉴로퍼지 제어기술은 신경회로망 구조에 퍼지 논리를 접목한 것으로 퍼지 법칙의 설정의 어려움을 신경회로망의 학습을 통해 해결하고자 하는 기술이다. 그 구조에 따라 다양한 방법이 있으며, T-S 방식의 뉴로-퍼지 제어기의 구조가 상대적으로 간단하여 많이 사용된다. 본 논문에서는 T-S 방식의 뉴로-퍼지 제어기를 사용하여 비선형 시스템인 도립역진자 제어를 수행하고자 한다.

기존의 연구를 통해서 신경회로망 제어기와 퍼지 제어기를 각각 사용하여 도립역진자 시스템의 수레의 위치 및 도립진자의 도립상태 제어의 가능성에 대해서 확인을 하였다[3-5]. 따라서, 본 논문에서는 신경회로망 제어기와 퍼지 제어기의 장점을 결합한 형태의 뉴로퍼지 제어기를 DSP와 FPGA를 결합한 제어 시스템을 하드웨어로 구현하고 도립역진자 시스템에 적용하여, 제어 성능을 PID제어기와 비교 평가하고자 한다.

2. 본 론

2.1 뉴로-퍼지

도립역진자 시스템과 같은 비선형 시스템을 제어하기 위한 퍼지 제어기의 성능은 작동자의 경험에 의한 퍼지 법칙을 얼마나 정확하게 생성하는가에 달려있다. 퍼지 제어를 위해서는 제어 변수와 이에 따른 제어 규칙의 설정, 소속함수 설정, 그리고 제어 변수의 정규화 과정이 필요하다. 퍼지 제어기를 실제로 적용 할 때 가장 문제가 되는 부분이 제어 규칙의 설정이다. 그러나 최적의 제어 규칙을 설계하는 일반적인 방법은 존재하지 않기 때문에 설계자의 경험과 지식에 크게 의존하게 되고, 이는 제어기의 설계 시간과 성능에 큰 영향을 미치게 된다. 또한 이미 작성한 퍼지 제어기는 시스템의 변화에 능동적으로 대처하기 어려운 단점이 있다. 즉, 시스템이 변화하게 되면 이에 맞춰 정규화 값은 변경해 주거나 제어기의 규칙을 수정해야 하는 번거로움을 안고 있다. 퍼지 법칙을 설계한 후에 시스템이 변화한다면 변화하기 이전의 시스템 기반으로 설계한 퍼지 법칙이 제대로 작동하지 않을 수 있다. 이 경우에 퍼지 법칙을 다시 구성해야 하는데 이는 매우 번거롭다. 따라서 시스템의 변화에 따라 자동적으로 법칙을 바꾸어주면 매우 편리하고 지능적이다. 이를 만족하기 위한 뉴로-퍼지 제어기의 목적은 퍼지 법칙을 상황에 따라 퍼지 법칙을 적용적으로 바꾸어 주기 위함이다. 퍼지 법칙을 바꾼다함은 퍼지 법칙을 구성하는 소속함수(Membership function)의 중심값과 폭을 들 수 있다. 기본적으로 퍼지 법칙을 나타내는 함수는 아래의 가우시안 함수를 많이 사용한다.

$$f(x) = \exp\left(-\left(\frac{x-\mu}{\sigma}\right)^2\right) \quad (1)$$

여기서 x 는 입력 데이터, μ 는 평균값으로 소속함수의 중심값을 나타내고 σ 는 분산으로 폭을 나타낸다. 이 두 변수를 바꾸어 주면 원하는 퍼지 법칙을 만들 수 있다.

아래 그림은 일반적인 뉴로퍼지 구조에서 비퍼지화 부분을 1차 선형식의 형태로 하는 Tagaki-Sugeno방식을 취하는 구조이다.

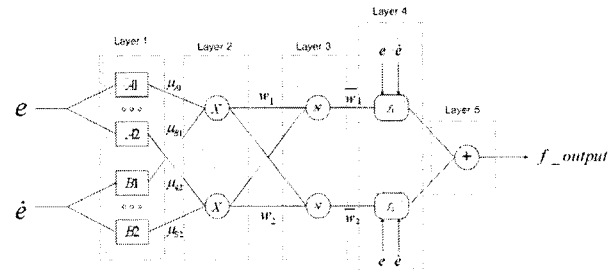


그림 1 T-S 뉴로-퍼지 제어기의 구조

Layer1에서는 각 입력의 소속함수의 소속도를 구하는 Layer로서 가오시안 함수를 사용할 경우 출력은 다음과 같다[6].

$$f_{11}^{(1)} = \mu_{A1} = G(m_{11}, \sigma_{11}) = \exp\left[-\left(\frac{e - m_{11}}{\sigma_{11}}\right)^2\right] \quad (2)$$

$$f_{12}^{(1)} = \mu_{A2} = G(m_{12}, \sigma_{12}) = \exp\left[-\left(\frac{e - m_{12}}{\sigma_{12}}\right)^2\right]$$

$$f_{21}^{(2)} = \mu_{B1} = G(m_{21}, \sigma_{21}) = \exp\left[-\left(\frac{\dot{e} - m_{21}}{\sigma_{21}}\right)^2\right]$$

$$f_{22}^{(2)} = \mu_{B2} = G(m_{22}, \sigma_{22}) = \exp\left[-\left(\frac{\dot{e} - m_{22}}{\sigma_{22}}\right)^2\right]$$

Layer2에서는 모든 소속 함수의 곱과 합을 생성한다.

$$\begin{aligned} f_1^{(2)} &= z_1 = \mu_{A1} \cdot \mu_{B1} \cdot \mu_{n1} \\ f_2^{(2)} &= z_2 = \mu_{A2} \cdot \mu_{B2} \cdot \mu_{n2} \\ b &= f_1^{(2)} + f_2^{(2)} \end{aligned} \quad (3)$$

Layer3에서는 나누기 연산을 수행한다.

$$f_1^{(3)} = \overline{w}_1 = \frac{f_1^{(2)}}{f_1^{(2)} + f_2^{(2)}}, \quad f_2^{(3)} = \overline{w}_2 = \frac{f_2^{(2)}}{f_1^{(2)} + f_2^{(2)}} \quad (4)$$

Layer4에서는 가중치와 곱을 실행하여 입력과 곱한다.

$$\begin{aligned} y_1 &= p_1 e + q_1 \dot{e} + r_1, \quad y_2 = p_2 e + q_2 \dot{e} + r_2 \\ f_1^{(4)} &= y_1 \overline{w}_1, \quad f_2^{(4)} = y_2 \overline{w}_2 \end{aligned} \quad (5)$$

Layer5에서는 합을 통해 최종 출력을 계산한다.

$$f^{(5)} = \sum_{i=1}^n f_i^{(4)} = f_1^{(4)} + f_2^{(4)} \quad (6)$$

2.2 학습 알고리즘

Tagaki-Sugeno 뉴로-퍼지 모델을 학습하기 위해 역전파 알고리즘을 적용한다. 현재 뉴로-퍼지에서 학습해야 할 변수들은 $\hat{p}, q, r, m_{ij}, \sigma_{ij}$ 이며, 최소화하고자 하는 목적 함수는 다음과 같다.

$$E = \frac{1}{2} \sum_{i=1}^n e_i^2 \quad (7)$$

여기서 오차 $e_i = f_d^p - f^p$ 로 각 패턴에 대한 오차이다. f_d^p 는 원하는 출력이고 f^p 는 뉴로퍼지의 출력이다. 각 변수를 학습하기 위해 gradient decent algorithm의 역전파 알고리즘을 사용한다.

$$\frac{\partial E}{\partial r_i} = -e \frac{z_i}{b}, \quad \frac{\partial E}{\partial p_i} = -e \frac{z_i}{b} x_1, \quad \frac{\partial E}{\partial q_i} = -e \frac{z_i}{b} x_2 \quad (8)$$

$$\frac{\partial E}{\partial m_{ij}} = -2e \frac{y_i}{b} \left(\frac{x_i - m_{ij}}{(\sigma_{ij})^2} \right) \prod_{i=1}^n \left[\exp\left(-\frac{(x_i - m_{ij})^2}{(\sigma_{ij})^2}\right) \right]$$

$$\frac{\partial E}{\partial \sigma_{ij}} = -2e \frac{y_i}{b} 2 \left(\frac{x_i - m_{ij}}{(\sigma_{ij})^3} \right) \prod_{i=1}^n \left[\exp\left(-\frac{(x_i - m_{ij})^2}{(\sigma_{ij})^2}\right) \right]$$

여기서 $e = f_d - f^{(5)}$, $x_1 = e$, $x_2 = \dot{e}$ 이다.

2.3 뉴로-퍼지 제어 방식

역진자 시스템은 하나의 입력으로 진자의 각도와 수레의 위치를 동시에 제어해야 하는 비선형 시스템이다. PID 제어를 사용하면 각도를 세울 수는 있으나 지속적으로 수레가 움직여 위치를 제어 할 수 없다. 또한 외란이 발생 했을 때 불안정해진다. 그러나, 비선형 제어기인 뉴로-퍼지 제어기를 사용하면 이 문제를 해결할 수 있다.

아래 그림 2는 본 논문에서 사용하고 있는 뉴로-퍼지 제어기의 구조를 나타낸다. 그림 2의 블록 다이어그램을 보면 알 수 있듯이, 진자의 각도 오차와 수레의 위치 오차에 대해 개별 뉴로-퍼지 제어기에서 제어 신호를 만들고, 그 결과를 더해서 Inverted Pendulum의 수레의 모터에 전달한다.

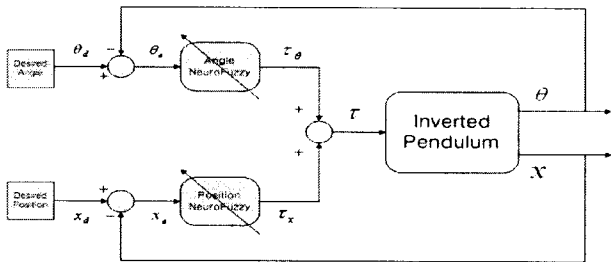


그림 2 뉴로-퍼지 제어 블록도

3. 실험

아래 그림 3과 그림 4는 뉴로-퍼지 제어기의 성능을 실험할 Inverted Pendulum 시스템과 제어기 연산을 수행할 DSP와 FPGA보드를 보여준다.

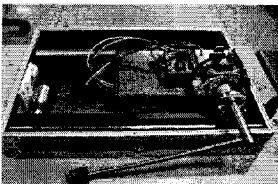


그림 3 Inverted Pendulum 그림 4 제어기 시스템

상기 시스템에서 PID 제어기와 뉴로-퍼지 제어기에 진자의 균형을 유지하는 상태에서 수레의 위치를 Sin파형태의 이동 경로가 주어졌을 때 진자의 각도 오차와 수레의 위치 오차를 비교 평가하는 방법으로 실험을 진행 하였다.

아래 그림 5는 PID 제어기를 사용했을 때 결과이며, 진자의 균형을 유지 하면서 수레가 Sin경로를 추종하고는 있으나, 많은 진동이 발생함을 알 수 있다. 흑색 실선은 수레의 현재 위치, 청색 실선은 수레의 위치 오차, 적색 실선은 진자의 각도 오차임 그림 6은 뉴로-퍼지 제어기를 사용했을 때 결과이며, 역시 진자의 균형을 유지하며 수레가 Sin경로를 잘 추종하고 있다. PID 제어기와 비교했을 때 진자의 각도 오차 및 수레의 위치 오차 역시 매우 개선되었음을 알 수 있다.

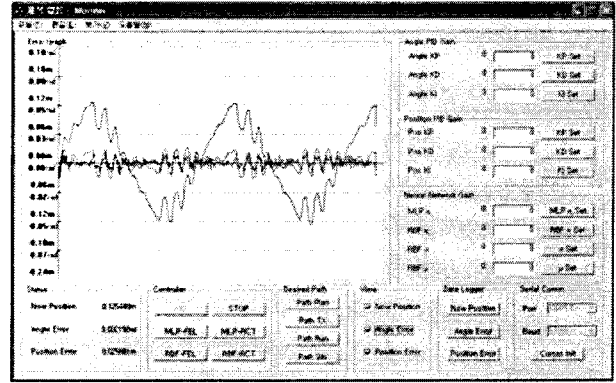


그림 5 PID 제어기 결과

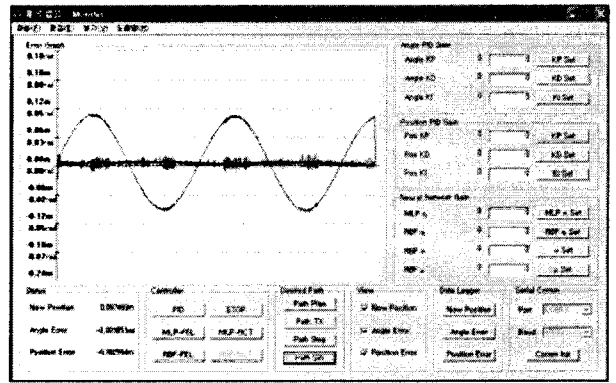


그림 6 뉴로-퍼지 제어기 결과

4. 결 론

본 논문에서는 뉴로-퍼지 제어 알고리즘을 DSP와 FPGA에 임베드하여 제어기를 설계하고 도립진자의 위치 추종 실험을 통해 뉴로-퍼지 제어기의 우수한 제어 성능을 PID 제어기와 비교해 증명하였다. 실험을 통해 퍼지 규칙을 변화하는 뉴로-퍼지제어기의 적응 능력을 알 수 있었다. 하지만, 초기에 무작위로 선택되었던 값들에 의한 불안정한 모습을 볼 수 있었다. 이는 뉴로-퍼지 제어기의 학습변수들이 시스템을 안정화 시킬 수 있을 때까지 안정화해주는 것이 필요한 것을 나타낸다. 추후에 이를 개선할 수 있는 방법에 대한 연구가 필요하다.

사 사

본 연구는 교육과학기술부와 한국산업기술재단의 지역혁신인력양성사업으로 수행된 연구결과이며 지원에 감사드립니다.

[참 고 문 헌]

- [1] M. E. Magana and F. Holzapfel(1998), "Fuzzy-logic control of an inverted pendulum with vision feedback", *IEEE Trans. on Education*, vol. 41, no. 2, pp. 165-170
- [2] I. Fantoni and R. Lozano(2003), "Global stabilization of the cart-pendulum system using saturation functions", *IEEE Conference on Decision and Control*, pp. 4393-4398
- [3] Cho, H. T., Jung, S. and Hsia, T. C. (2007) "Neural network control for position tracking of a two-axis inverted pendulum system: Experimental studies", *IEEE Transaction on Neural Networks*, vol. 18, no.4, pp.1042-1048.
- [4] Geun Hyeong Lee, Jin seok Noh, Seul Jung(2007), Jung, S. and Cho, H. T.(2004) "Implementation and experiment of neural network controllers for intelligent control system education", *International Journal of Fuzzy logic and intelligent systems*, pp. 267-273
- [5] Deok Hui Song, Geun Hyeong Lee, Seul Jung(2008), "Neural network compensation technique for standard PD-like fuzzy controlled nonlinear systems", *International Journal of Fuzzy logic and intelligent systems*, pp.68-74
- [6] 정슬, "인공지능시스템 II, 지능제어", 충남대학교 출판부, 2005