

상대이득 행렬 기법을 이용한 신경망 제어기 설계에 관한 연구

서호준*, 서삼준**, 김동식***, 박귀태*

*고려대학교 전기공학과 **안양대학교 전기공학과 ***순천향대학교 제어계측공학과

A Study on The Neural Network Controller using Relative Gain Matrix Technique

Seo Hojoon*, Seo Samjun**, Kim Dongsik***, Park Gwitae*

*Korea Univ. Electrical Eng. **Anyang Univ. Electrical Eng. ***SoonChunHyang Univ. Control & Instr.

Abstract - In this paper, Neuro-Fuzzy Controller(NFC), a fuzzy system realized using a neural network, is to adopt for the multivariable system. In the multivariable system, the interactive effects between the variables should be taken into account. A simple compensator, using the steady-state information can be obtained for open-loop stable systems, is presented to cope with this problem. However, it should be supposed that the plant is unknown to the control system designer, but an estimate of the DC gain has been obtained by carrying out experiments on the plant. Also, if the variables are not combined completely, it is difficult to design the controller. Therefore, we design a neuro-fuzzy controller which controls a multivariable system with only input output informations, and compare its performance with that of a PI controller. In the proposed controller, the construction of the membership functions and rule base, which is highly heuristic, can be achieved using a training process. This allows the combination of knowledge of human experts and evidence from input-output data.

1. 서 론

다변수 계통에서는 제어 입력과 상태 변수간의 강한 결합성이 제어성능에 큰 영향을 미치게 된다. 특히 계통이 비선형일 경우에는 결합성으로 인한 상호 영향이 계통을 제어하기 더욱 힘들게 하는 한 요인이 된다. 이와 같은 결합성이 강한 다변수 계통을 효과적으로 제어하기 위하여 여러 가지 해석적인 기법들이 있다. 60년대 중반 Bristol에 의하여 처음 제안된 상대이득행렬(RGM)을 이용하여 강한 영향을 주는 변수끼리 페어링함으로써 감결합효과를 얻는 방법이 있다.[3] 이 방법은 계통의 정상상태 정보만으로 제어기 출력변수와 제어 상태변수

(controlled variable)간의 영향 정도가 강한 것끼리 1대 1로 짝을 지어 우세한 짝끼리 제어하는 기법으로 최근까지 초기의 개념을 변형시킨 여러 가지 기법이 효과적으로 사용되고 있다. 그러나 상대이득행렬을 이용한 퍼지 논리 제어기에서는 알려지지 않은 플랜트에 대해서 실험을 통한 정상상태이득의 추정치를 알아야만 한다. 또한 변수들의 상호작용에 대한 결합이 잘 이루어지지 않는 경우에 대해서는 제어기를 설계하기 어렵다. 따라서 본 논문에서는 입출력 정보만을 가지고 다변수 계통을 제어하는 뉴로 퍼지 제어기(NFC) 구조를 설계하고 또 이를 PI제어와 비교하여 성능을 비교하도록 한다.

2. 다변수 퍼지 논리 제어기

일반적으로 입력변수가 n 개이고 출력변수가 m 개인 다변수 퍼지 논리 제어기(MIMO 퍼지제어기)는 입력변수가 n 개이고 출력변수가 한 개인 MISO 퍼지제어기의 결합으로 나타낼 수 있다.[5-6] 그러나 다변수 계통에서는 제어 입력과 상태 변수간의 강한 결합성이 제어성능에 큰 영향을 미치게 된다. 이와 같은 결합성이 강한 다변수 계통을 효과적으로 제어하기 위하여 60년대 중반 Bristol에 의하여 처음 제안된 상대이득행렬(RGM)을 이용하여 강한 영향을 주는 변수끼리 페어링을 함으로써 감결합효과를 얻는 방법을 간단하게 설명한다.

다변수 프로세서로부터 야기되는 어려움을 조절하기 위해 다변수 퍼지제어기를 다음과 같이 구성하였다.[8]

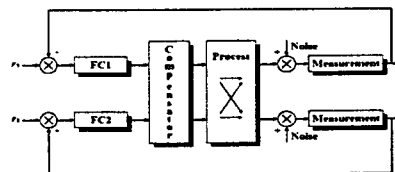


그림 1 다변수 퍼지 제어기의 구조

다변수 시스템에서 정상상태이득들은 행렬 G 에 의해 표시되어진다. 대부분의 경우에, G 는 고려되어진 시스템에 대한 실험을 통해 쉽게 얻어질 수 있다. 반면, 만약 프로세서가 미분방정식의 집합으로 특징화되어진다면, G 는 다음의 절차를 통해 유도되어질 수 있다.

시스템이 다음과 같이 주어진다고 가정하자.

$$\begin{aligned} \frac{dx}{dt} &= f(x, u) \\ y &= Cx \end{aligned} \quad (1)$$

여기서 f 는 함수벡터, $x \in \mathbb{R}^n$ 는 상태벡터, $u \in \mathbb{R}^m$ 는 제어벡터, $y \in \mathbb{R}^m$ 는 출력벡터, 그리고 C 는 $m \times n$ 상수행렬이다. 미소 상태를 근처에서의 증분 모델은 테일러시리즈의 1차 근사화에 의해 다음과 같이 얻어진다.

$$\begin{aligned} \frac{d\Delta x}{dt} &= A\Delta x + B\Delta u \\ \Delta y &= C\Delta x \end{aligned} \quad (2)$$

입력 u_j 와 출력 y_i 사이의 상대이득 γ_{ij} 는 두 개의 정상상태 이득들의 비이다. 즉 모든 루프들이 개방되었을 때 u_j 와 y_i 사이의 이득 g_{ij} 와 i 번째 출력을 제외하고 모든 출력들이 빈틈없이 제어되어졌을 때의 이득 \hat{g}_{ij} 의 비이다.

상대이득행렬은 단지 정상상태이득행렬 G 를 사용함에 의해 계산되어질 수 있다. 상대이득행렬 Γ 의 (i, j) 번째 요소 γ_{ij} 는 다음과 같이 표현되어진다.

$$\gamma_{ij} = g_{ij} \cdot \hat{g}_{ij} \quad (3)$$

상호작용의 측정으로서 소개된 상대이득행렬은 입출력 변수들의 결합을 위해, 또 상호작용이 중요하게 되었을 때 예견하기 위해 지침들을 제공한다. 목표는 상대이득이 양수이고 가능한 한 1에 가까운 변수들을 결합함에 의해 달성될 수 있다. 만약 γ_{ij} 가 1보다 크거나 0보다 작다면, 대응하는 변수들의 결합은 제어하기 어려운 루프가 될 것이다.

3 제안된 뉴로 퍼지 제어기

상대이득행렬의 계산으로부터 입출력데이터의 페어링을 이용할 수 있는 경우에 그림 4.1과 같이 보상기를 달아서 decoupling 효과를 보고자 한다.

그림 2는 입출력 페어링이 $u_1 \leftrightarrow y_1, u_2 \leftrightarrow y_2$ 와 같이 된 경우이다. 이때, y_1 에 대한 u_2 의 영향과 y_2 에 대한 u_1 의 영향을 고려해주기 위해 보상기를 인가하며, 이 보상기의 변수들은 신경회로망에 의해 오차를 최소화하는 방향으로 학습된다. 따라서 상대이득행렬의 추정치를 이용할 수 있는 경우 각 입

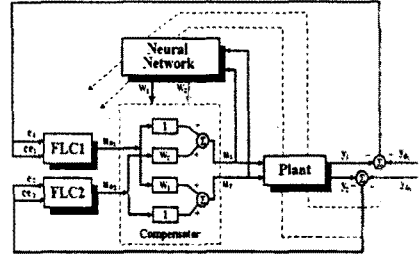


그림 2 제안된 뉴로 퍼지 제어기

출력간의 결합(coupling)을 고려하지 않고 오직 FLC의 강인성만 가지고 보상하게 되었을 때 결합이 강한 프로세스의 경우 제어성능이 저하되는 문제점을 이와같이 보상기를 인가하여 더 나은 성능을 가질 수 있도록 하였다.

제어입력은 식 (4)와 같다.

$$\begin{aligned} u_1 &= 1 \times u_{a1} + W_2 \times u_{a2} \\ u_2 &= W_1 \times u_{a1} + 1 \times u_{a1} \end{aligned} \quad (4)$$

역전과 학습 알고리즘을 통해 신경회로망의 연결강도의 변화량을 구하면 오차 함수는 식 (5)와 같다.

$$\begin{aligned} E &= \frac{1}{2} (e_1^2 + e_2^2) \\ &= \frac{1}{2} \{ (y_{a1} - y_1)^2 + (y_{a2} - y_2)^2 \} \end{aligned} \quad (5)$$

출력단의 연결강도의 변화량은 식 (6)과 같다.

$$\begin{aligned} \Delta w_{kj}(n+1) &= -\eta \frac{\partial E}{\partial w_{kj}} \\ &\quad + \alpha \Delta w_{kj}(n) + \beta \Delta w_{kj}(n-1) \end{aligned} \quad (6)$$

연쇄법칙을 이용해서 오차함수 식 (5)에서 연결강도 변화량 $\partial E / \partial w_{kj}$ 를 구하면 식 (7)과 같다.

$$\frac{\partial E}{\partial w_{kj}} = \frac{\partial E}{\partial y} \frac{\partial y}{\partial u} \frac{\partial u}{\partial O_k} \frac{\partial O_k}{\partial net_k} \frac{\partial net_k}{\partial w_{kj}} \quad (7)$$

여기에서,

$$\frac{\partial E}{\partial y} = \left[\frac{\partial E}{\partial y_1} \quad \frac{\partial E}{\partial y_2} \right] = [-e_1 \quad -e_2]$$

$$\frac{\partial y}{\partial u} = \begin{bmatrix} \text{sign}\left(\frac{\partial y_1}{\partial u_1}\right) & \text{sign}\left(\frac{\partial y_1}{\partial u_2}\right) \\ \text{sign}\left(\frac{\partial y_2}{\partial u_1}\right) & \text{sign}\left(\frac{\partial y_2}{\partial u_2}\right) \end{bmatrix} \quad (8)$$

$$\begin{bmatrix} \frac{\partial u_1}{\partial O_1} \\ \frac{\partial u_2}{\partial O_1} \end{bmatrix} = \begin{bmatrix} u_{a2} \\ 0 \end{bmatrix}, \quad \begin{bmatrix} \frac{\partial u_1}{\partial O_2} \\ \frac{\partial u_2}{\partial O_2} \end{bmatrix} = \begin{bmatrix} 0 \\ u_{a1} \end{bmatrix}$$

여기서, O_1 는 신경회로망의 출력으로 W_1 과 같으며, O_2 는 W_2 와 같다.

식 (6)과 식 (7)에 의하여 신경회로망의 연결강도의 변화량을 구할 수 있고 이로인해 학습이 가능하다. 전체적인 학습 알고리즘은 다음과 같다.

- [1단계] 연결강도와 바이어스 값을 0으로 초기화
- [2단계] 플랜트 제어입력을 신경망 입력으로 인가
- [3단계] 목표값과 실제 출력값사이의 오차 획득
- [4단계] 연결강도 변화

▶ 출력층 연결 강도의 변화량 결정

$$\Delta w_{kj}(n+1) = -\eta \cdot \delta_k \cdot O_j + \alpha \Delta w_{kj}(n) + \beta \Delta w_{kj}(n-1) \quad (9)$$

$$\delta_k = [-e_1 - e_2] \cdot \text{sign}\left[\frac{\partial y}{\partial u}\right] \cdot \frac{\partial u}{\partial O_k} O_k(1-O_k)$$

▶ 중간층 연결 강도의 변화량 결정

$$\Delta w_{ji}(n+1) = -\eta \cdot \delta_j \cdot O_i + \alpha \Delta w_{ji}(n) + \beta \Delta w_{ji}(n-1) \quad (10)$$

$$\delta_j = O_j(1-O_j) \sum_k \delta_k w_{kj}(n)$$

- [5단계] 보상기로 인가하는 연결강도 획득
- [6단계] 2단계에서 반복

4. 컴퓨터 시뮬레이션 및 결과

대상으로 하는 계통은 다음과 같다.

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -2 & -4 & -3 \end{bmatrix} x + \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -1 & 1 \end{bmatrix} u \quad (11)$$

$$y = \begin{bmatrix} 0 & 1 & -1 \\ 1 & 2 & 1 \end{bmatrix} x \quad (12)$$

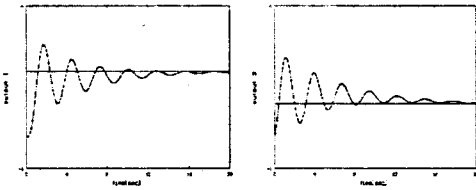


그림 4 PI제어기에 의한 출력

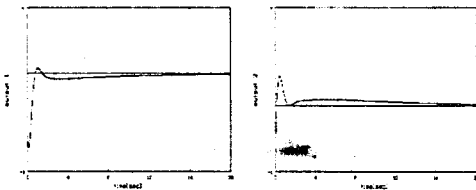


그림 5 제안된 제어기의 출력

6. 결론

본 논문에서는 퍼지와 신경망의 장점을 갖는 뉴로 퍼지 제어기를 설계하여 이를 다변수 계통에 적용하였다. 다변수 계통은 제어 입력과 상태 변수간의 결합성으로 인한 상호영향을 고려하기 위해서 상대이득행렬을 이용한 퍼지 논리 제어기로 MIMO시스템에 적용하는 경우에 대해 살펴보았다. 제안된 제어기는 입출력정보만을 이용하여 학습을 통해 이러한 상호영향에 대한 고려를 하지 않고 원하는 출력을 쫓아가도록 뉴로 퍼지 제어기를 구성하였으며, 이에 따라 자동적으로 소속함수값과 퍼지 규칙을 찾아가도록 하였다.

참고 문헌

- [1] Ryusuke Masuoka, Nobue Watanabe, "Neurofuzzy system-fuzzy inference using a structured neural network", Proceedings of the International conference on Fuzzy Logic & Neural Networks, pp.173~177, July, 1990
- [2] Alberico Menozzi, Mo-yuen Chow, "A Design Methodology for an Intelligent Controller using Fuzzy Logic and Artificial Neural Networks", IEEE, pp.408~413, 1993
- [3] Grosdidier, P. and Morari, M, "Closed-loop properties from steady-state gain information.", Industrial and Engineering Chemistry Fundamentals, 24, pp.221-235, 1985
- [4] 김대수, "신경회로망과 응용(I)", 하이테크정보, August, 1992
- [5] 채석, 오영석, "퍼지이론과 제어", 청문각, August, 1995
- [6] Madan M. Gupta, "Multivariable Structure of Fuzzy Control Systems", IEEE Trans. Sys., Man, Cyber., Vol. 16, No.5, pp.638-655, 1986
- [7] Daniel E. Miller, Edward J. Davison, "The Self-Tuning Robust Servomechanism Problem.", IEEE Transactions on Automatic Control, Vol.34, NO.5, pp 511-517, 1989
- [8] Junhong Nie & Derek A. Linkens, "Fuzzy-neural control principles, algorithms and applications", prentice Hall, pp37~67, 1995
- [9] Shin-ichi Horikawa, Takeshi Furuhashi, "On Fuzzy Modeling Using Fuzzy Neural Network with the Back-propagation Algorithm", IEEE Transactions on neural networks, Vol.3, NO.5, pp.801~806, 1992
- [10] Hashiyama, "An Interval Fuzzy Model Using a Fuzzy Neural Network", IEEE, pp.745~750, 1992