

신경망에 기초한 T-S 퍼지 규칙의 자동생성과 표현^{*}

Automatic Generations and Representations of T-S Fuzzy Rule based on Neural Networks

황 문 선 · 오 경 환

Hwang Moon-sun and Oh Kyung-whan

서강대학교 컴퓨터학과

요 약

본 논문에서는 기존의 퍼지 제어 규칙에 비해 좋은 성능을 갖는 T-S(Takagi-Sugeno)퍼지 모델을 자기조직화 지도와 역전파 신경망을 이용하여 표현하고, 제어기 구현을 위한 규칙의 자동 생성 방법을 제안한다. 제안된 방법은 신경망에 기초하여 T-S 퍼지 제어 규칙을 표현하므로써 학습 기능을 이용하여 지식 획득을 용이하게 하고, 입력 변수간의 퍼지 관계에 기반 하여 추론이 이루어지므로 각 퍼지 변수에 대한 소속 함수의 정의 과정이 불필요하게 된다. 또한 제어기로 구현 되었을 때 규칙의 수나 퍼지화 및 비퍼지화 등이 구성된 추론망을 통하여 자동으로 수행될 수 있다. 때문에 퍼지 시스템의 구현이 쉽게 이루어 질 수 있게 한다. 제안된 방법을 자동차 궤도 안정화 모의 실험에 적용해 봄으로써 추론망이 규칙을 생성하여 타당한 추론을 하게 됨을 확인한다.

ABSTRACT

In this paper, we present T-S(Takagi-Sugeno) fuzzy model which shows better performance than traditional fuzzy control rules by self-organizing map and back propagation neural network, and show a network, inference network, that automatically generates fuzzy rules for designing controller, Proposed network's learning ability makes knowledge acquisition much easier. Because fuzzy inference is based on fuzzy relations in our approach, the membership functions of each variable is not needed. In controller, the number of rules, fuzzification, and defuzzification is mechanically performed through the inference network. Therefore the implementation of fuzzy controller become simple and we will certify our network through trajectory stabilization of a model car.

I. 서론

퍼지 제어는 경험적인 제어지식을 퍼지 규칙의 형태로 기술하고 퍼지 추론에 의해 제어동작을

결정하는 방법으로, 모델의 구조가 이해하기 쉽고, 비선형적인 입·출력 관계에도 용이하게 표현될 수 있는 특징이 있다. 대부분의 퍼지 제어 시스템은 조작자의 경험이나 엔지니어의 지식으로부터 얻은 규칙들로 이루어진 규칙 기반 시스템이다. 그러나 시스템의 복잡도가 높아질수록 조작자의 행동을 표현하는 것이 어렵고, 지식으로부터 작성된 규칙들은 실제의 입출력 관계를 표현할 수 없는 경우가 있어서 규칙의 조정이 필요하다. 퍼지 제어 시스템의 구현에 있어서 지식획득의 문제점의 해결방법 중의 하나는 조작자의 행동에 따른 입출력 자료를 근거로 조절 행위의 퍼지 모델을 만드는 것이다. 신경망의 특징은 분산 표현과 학습 능력으로서 비선형적인 관계를 결정할 수 있는데, 이는 신경망이 퍼지 시스템에의 문제점을 해결하는 방법으로 사용될 수 있도록 하여준다.

이와 같이 퍼지 모델과 신경망 모델을 결합하여 각 모델의 문제점을 해결하고 보다 나은 성능을 갖는 시스템을 만들려는 다양한 방법이 제안되어 왔는데[2-4], 본 논문에서는 Takagi와 Sugeno가 제안한 퍼지 모델(T-S fuzzy model)[5]을 신경망으로 표현하였다. T-S 퍼지 모델은 기존의 퍼지 모델이 많은 퍼지 변수를 가지게 될 때의 문제점을 해결하고자 제안된 모델로서 퍼지 제어를 구현하는데 많이 사용되고 있다. 이를 신경망으로 표현하기 위해서 비교사 학습 신경망인 자기 조직화 지도(self-organizing map: SOM)와 교사 학습 신경망인 역전파 신경망의 두 가지 신경망이 사용되는데, 각각이 조건부와 결론부를 대신하고 있다. 조건부 신경망에서는 입력 값의 퍼지화와 소속 함수의 결정, 생성된 규칙에 대한 소속 정도 등을 자동으로 결정하여 주고, 결론부 신경망에서는 T-S 퍼지 규칙의 결론부 함수 값을 계산하게 되며, 이 두 가지 신경망을 서로 연결하여 각각의 결과를 바탕으로 비퍼지화를 수행하므로써 원하는 제어 결과를 얻게 된다.

이러한 과정을 통해서 퍼지 집합과 소속 함수의 정의가 생략될 수 있고 규칙에 대한 소속 함수를 쉽게 구할 수 있게 되어 퍼지 시스템의 구현 과정이 간단해지며, 신경망의 병렬 수행 능력과 조건부, 결론부 신경망의 병렬 연결로 인하여 신속한 추론을 가능하게 된다. 또한 제안된 추론망은 조건부, 결론부, 비퍼지화의 구조로 T-S 퍼지 규칙을 명확하게 표현함을 알 수 있게 된다.

II. Takagi-Sugeno 퍼지 모델

어떤 상황에 조작자가 행하는 제어 행위는 수치적으로 표현이 불가능한 경우가 흔히 있다. 이러한 문제는 수치적인 자료를 사용하여 조작자의 행위는 모델링하게 되는데, 이 경우 역시 많은 수의 규칙이 필요하여 많은 시간이 소요된다는 문제점을 가진다. 이러한 문제를 해결하기 위해 다음과 같은 형태의 규칙을 사용하는 것이 가능하다.

$$R_i : \text{IF } (x_1 \text{ is } T_{x_1} \text{ and } \dots x_n \text{ is } T_{x_n}) \text{ THEN } g(i) = p_0 + p_1x_1 + \dots + p_nx_n$$

Takagi와 Sugeno에 의해 제안된 이 모델[5]은 일반적인 퍼지규칙과는 달리 제어기의 결론부분의 입력 값이 x_i 의 선형함수로 정의되어 있다. 따라서 결론부를 위한 소속함수는 필요 없다.

결론부 함수의 계수 p_i 는 제어기의 성능을 최적화하기 위한 매개변수인데, 함수의 근사화 과정으로 주어진 데이터로부터 추정된다. 결론부 함수의 출력이 규칙이 결론이 되기 때문에 T-S 퍼지 제어 모델은 일반적인 퍼지 제어 모델과 비퍼지화 과정을 수행하는 것이 다르다.

퍼지 시스템을 구현하기 위한 규칙 생성의 방법에는 전문가의 지식을 도입하거나, 인간의 제어 기능을 추출하거나, 공정의 퍼지 모델을 이용하는 방법이 있다. 이들을 각각 도입되는 형태와 추론에서의 논리가 다른데, T-S 퍼지 규칙은 조작자의 제어 기능을 추출하여 모델링하는데 효과적이다.

T-S 퍼지 규칙을 표현하기 위해서는 제어기에서 입력 값들을 퍼지화시켜 규칙에 대한 소속 함

수를 정의하는 것과 결론부의 함수의 계수를 결정하는 것이 중요하다. 고전적인 퍼지 규칙과는 달리 규칙의 소속정도를 조건부에서 바로 구할 수 있고, 결론부에서의 함수는 필요 없다. 따라서 여기서 비퍼지화는 조건부의 규칙의 정의와 결론부의 결과 값을 모든 함수에 수행한 후 이루어 질 수 있다.

III. 추론망의 구축

3.1 조건부 신경망의 구축

조건부를 구성하는 신경망은 3개의 층으로 각층은 다음과 같다. 첫 번째 층에서는 입력을 받아 입력 값들이 궁극적으로 되어야 하는 상태와의 차이를 구하고, 두 번째 층에서는 입력된 값과 적당한 본포 정도를 출력하게 된다. 세 번째 층은 이러한 각 퍼지 값의 소속 정도를 이동하여 입력된 값이 전체 규칙에 대해 각각 어느 정도의 소속 정도를 가지게 되는 지를 구하게 된다.

학습을 시키기 위한 조건부 신경망의 입력 값들은 제어 시스템에서 사용될 때와는 다른 데이터가 사용된다. 일단 학습이 끝난 후 시스템에 적용될 때의 입력 값들은 제어할 상황의 값들이지만, 조건부 신경망을 학습시키기 위한 입력 값들은 제어할 상황의 값들이 발생할 수 있는 전체 범위에서 임의로 선택한 입력 값으로 학습을 시키게 된다. 학습을 위해 임의로 선택된 입력 가능 범위의 값들은 정규화 되어 첫 번째 층으로 입력이 되게 되는데, 궁극적으로 되어야 하는 값과의 차이가 구해지게 된다.

자기조직화지도 신경망에서는 서로 위치적으로 가까운 입력 데이터들을 같은 군집으로 묶는 역할을 한다. 만약 k 개의 군집으로 나누었다면, k 번째 뉴런과 연결된 연결 강도 벡터는 입력 공간상에서 k 번째 군집의 중심점을 갖고 있고, 하나의 데이터에 대해서 군집에 속하는 정도를 결정할 수 있다. 이런 특징을 이용하여 k 번째 군집에 대한 R_k 의 퍼지 관계식을 정의한다. 입력 벡터 X 와 연결 강도 벡터 W 와의 거리의 차이가 크다면 입력 벡터 X 가 W 가 대표하는 군집에 속하는 정도가 작다는 것이고, 거리 차가 작다면 반대의 의미가 된다. 따라서 퍼지 관계식 $\mu_{R_k}(X)$ 는 아래의 식과 같이 정의된다. ($1 \leq r < 2$)

$$\mu_{R_k} = \exp(-2e(x_i - w_{ki})^2 / \sigma_{ki}^2), (k=1, 2, \dots, K) \left(\begin{array}{l} x_i = \text{input} \\ w_{ki} = \text{weight} \\ \sigma_{ki} = \frac{|W_{ki} - W_{closest}|}{r} \end{array} \right)$$

여기서 $w_{closest}$ 는 w_{ki} 와 가장 인접해 있는 뉴런의 연결 강도이다. 따라서 σ_{ki} 는 군집의 유효한 폭을 나타내며, $\mu_{R_k}(X)$ 에 의해 군집의 중심 w_{ki} 에서는 '1', w_{ki} 에서 σ_{ki} 보다 멀리 떨어져 있는 입력 값에 대해서는 k 번째 군집에 대한 소속 정도가 '0'이 된다. 조건부 신경망의 첫 번째와 두 번째 층은 입력 변수의 개수만큼의 1차원 SOM으로 구성된다. SOM의 학습이 완료되면 처음 두 개의 층을 지나면서 퍼지화를 자동으로 수행할 수 있게 된다.

SOM을 지나오면서 각 입력 값은 1차 소속 함수에 의해 퍼지값으로 변환이 되고, 이 퍼지값들은 세 번째 층으로 보내어진다. 세 번째 층에서는 입력의 퍼지값을 사용하여 각 규칙의 소속 정도를 구하게 된다. 세 번째 층의 뉴런의 수는 규칙의 수와 같고, 따라서 하나의 뉴런이 하나의 규칙에 대한 소속 정도를 출력한다. 세 번째 층의 뉴런들은 퍼지 AND를 수행하는 퍼지 뉴런들로서, 두 번째 층과 각 SOM의 뉴런들의 조합으로 연결되어 있다. 그러므로 모든 규칙에 대해 병렬적으로 입력 값의 소속 정도가 계산되어 진다.

3.2 결론부 신경망의 학습

T-S 퍼지 규칙은 결론부를 조건부 변수의 함수 형식으로 나타낸다. 따라서 결론부에서는 소속 함수의 정의가 필요하지 않다. 대신 T-S 모델은 매개변수 식별문제가 선행되어야 하는 어려움을 가지고 있지만, 신경망을 이용하므로써 매개변수 식별 문제를 제거한다.

결론부 함수는 기대값이 주어지는 교사 학습을 하기 때문에 결론부 신경망만을 학습시키려면 각 규칙의 결론부 함수 값에 대한 기대값이 필요로 하게 된다. 그러므로 각 규칙에 대한 기대값 대신 전체 제어기의 출력 값을 기대값으로 하는 시스템을 구현하므로써, 학습 데이터를 쉽게 얻을 수 있다. 제어기의 출력 결과를 기대값으로 하는 것은 퍼지 시스템의 경우 이미 비퍼지화를 수행한 후의 단일 값이므로, 이를 사용하여 학습을 수행하기 위해서는 역으로 단일 값에서 각 규칙에의 결론부 함수의 기대값을 구하는 과정이 선행되어야 한다. 이러한 문제점은 비퍼지화 과정이 포함되어 있는 전체적인 추론망을 학습시키므로써 해결할 수 있다.

결론부 함수의 근사화를 위한 전체 추론망의 구조는 비퍼지화 부분이 하나의 층을 차지하고, 조건부 신경망과 결론부 신경망은 비퍼지화를 위해 이 층에서 서로 연결되는데, 조건부 신경망과 비퍼지화 층의 연결을 위한 부가적인 층을 전체 규칙에 대한 각 규칙의 참여 정도를 계산하기 위한 층이다. 입력 데이터들은 추론망에 의하여 비퍼지화까지 수행된 단일 값으로 계산되어 제어기의 기대값과 비교되어 짐으로써 학습을 하게 된다. 결론부 함수의 근사화는 크게 전방향 전파와 역방향 전파로 구성된다.

전방향 전파는 P개의 학습 데이터 중에서 x_{pk} 가 입력으로 주어지면, x_{pk} 는 조건부 신경망과 결론부 신경망으로 보내어진다. 여기서는 비퍼지화 과정으로 무게 중심법을 사용한다. 비퍼지화를 위한 식은 다음과 같다. u_k 는 조건부 신경망 k번째 규칙의 출력, 결론부 신경망에서 O_{pk} 는 k번째 규칙의 출력을 w_k^Y 은 k번째 뉴런과의 연결 강도를 나타낸다.

$$\xi_k = \frac{\mu_k}{\sum_{r=1}^M \mu_r}, \quad Y_p = \sum_{k=1}^M O_{pk} w_k^Y \xi_k$$

u_k 는 비퍼지화층 전의 부가적인 층에서 조건부 적합도 ξ_k 로 계산되고, ξ_k 와 O_{pk} 는 p번째 입력 데이터 X_p 에 대한 단일 값인 결과 Y_p 를 얻는다.

전방향 전파를 통해 얻어진 결과 Y_p 는 역전파 학습의 목적 함수에 사용되어 역방향으로 학습이 시작된다. 여기서는 다음과 같은 목적 함수를 사용한다.

$$\delta_p = (D_p - Y_p), \quad E_p = \frac{1}{2} \delta_p^2$$

다음은 결론부 신경망의 각층에서 연결강도를 수정하기 위한 식들이다.

$$\Delta w_k^Y = \eta Y E_p^Y \xi_k O_{pk}$$

$$\Delta w_{kj}^O = \eta O E_{pk}^O \xi_k O_{pj}$$

$$\Delta w_{ji}^I = \eta I E_{pj}^I \left(\sum_{k=1}^M \xi_k \right) x_{pi}$$

$$Y E_p^Y = \delta_p, \quad O E_{pk}^O = Y E_p^Y w_k^Y f'(O_{pk}), \quad I E_{pj}^I = \sum_{k=1}^M O E_{pk}^O w_{kj}^O f'(I_{pj})$$

$$f(x) = \frac{1}{1 + e^{-x}}, \quad f(x) = (1 - f(x))f(x)$$

결론부 신경망에서 사용된 $f(x)$ 는 시그모이드형 함수이다. 학습은 전체 학습 데이터에 대하여 $E_p < \theta$ 이거나 반복횟수가 제한 횟수보다 많을 때까지 사용한다.

모든 학습과정을 거쳐 생성된 퍼지 시스템의 퍼지 규칙은 실질적으로 추론망의 연결 강도에 의해 표시된다. 이러한 이유로 구축된 추론망은 퍼지 시스템의 추론 엔진 역할뿐만 아니라 퍼지 규칙 베이스의 역할을 함께 한다. 그러므로 기존의 퍼지 논리 시스템에 순차적으로 이루어지는 규칙 베이스의 탐색이 병렬적으로 이루어 질 수 있으며, 규칙 베이스 탐색 과정을 거친 후 별도로 이루어지던 추론 과정이 탐색과 동시에 이루어지게 되므로 실시간 처리가 필요한 시스템에 유용하다. 또한 가능한 입력 데이터 내의 변수간의 퍼지 관계를 기반으로 하여 추론이 이루어지므로 데이터의 모든 변수에 대한 퍼지 소속 함수를 따로 정의하는 대신 하나의 퍼지관계식만을 정의하면 된다. 따라서 지금까지 시스템 구현 과정에서 퍼지 변수의 소속 함수 모양 등을 조정해야하는 과정이 생략될 수 있으므로 퍼지 시스템의 구현이 수월해진다고 볼 수 있다. 물론 퍼지 관계 함수의 정의도 잘 이루어져야 하고, 함수의 조절 등이 필요한 경우가 발생할 수 있으나, 기존의 퍼지 시스템에서 많은 퍼지 변수에 대한 소속 함수의 조정을 하는 것보다는 훨씬 쉽게 퍼지 시스템 구현이 가능하다.

IV. 실험 및 결과

지금까지 살펴본 추론망이 실제 제어 문제에 적용되어 규칙을 생성하고 타당한 제어 결과를 출력하는지 보기 위해서 자동차 궤도 안정화 문제에 적용하여 본다.



그림 1 모델 자동차와 제어시스템

자동차의 바퀴 각도 제어는 바퀴 각도에 대한 사람의 수학적인 모델을 모르기 때문에 경험적 지식만이 필요하므로 퍼지제어에 적합한 모델이다. 자동차가 임의의 위치에 임의의 각도를 있을 때 목표 위치에 진입하도록 자동차를 움직이는 것이다. <그림 1>은 이 모델을 보여 주고 있는데, 자동차는 전·후진이 모두 가능하지만 여기서는 복잡한 제어를 위해서 전진만이 가능하도록 한다. x_0 는 자동차의 초기 속도, x_1 는 자동차 후면의 수직 위치, u 는 앞바퀴의 각도, 길이 $l=2.8(m)$, 속도 $v=1.0(m/sec)$, 제어 간격 $t=1.0(sec)$ 을 사용하였다.

제어 시스템은 자동차의 현재위치 $x(k)$ 와 제어된 바퀴 각도 $u(k)$ 를 입력 값으로 다음위치 $x(k+1)$ 을 출력한다. 퍼지 신경망 제어기 부분은 제안된 추론망으로 구성된 제어기이다. 현재위치 (x_0, x_1) 를 입력으로 하면, 조건부 신경망에서 x_0 에 대한 뉴런이 7개, x_1 에 대해 5개, 따라서 총 규칙의 수는 35개가 된다. 이때 출력층의 뉴런의 개수는 35개, 은닉층은 20개로 정한다.

구축된 신경망을 직접 학습시킨 결과는 위의 그림과 같이 보여진다. 결론부 함수를 근사화하기 위한 결론부 신경망을 학습 데이터에 대해 바퀴 각도의 오차가 1도 이하가 되도록 학습시켰다.

제안된 추론망은 비교적 빠른 학습이 이루어 졌는데 이는 결론부 신경망의 출력층 뉴런들의 입력 값에 대하여 해당하는 규칙 뉴런들만 학습하도록 구성되었기 때문이다.

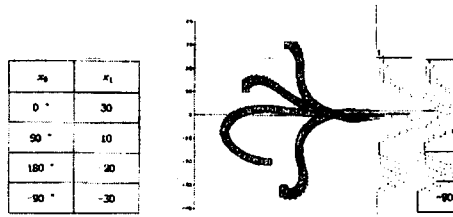


그림 2 모의 실험의 결과

V. 결론

본 논문에서는 일반적으로 퍼지 시스템을 구현하는 과정에 있어서 어려운 과정으로 여겨지는 지식 획득과 퍼지 변수의 소속 함수 정의 과정이 갖고 있는 문제점을 해결하는 신경망에 기초한 퍼지 추론망을 제안하였다. 제안한 방법은 제어를 위해 기존의 퍼지 모델의 기능을 향상시킨 T-S 퍼지 모델을 SOM과 역전파 신경망을 사용하여 조건부, 결론부, 비퍼지화 부분으로 명확히 표현한다. 신경망으로 표현된 T-S 퍼지 규칙의 추론망을 사용하므로써 제어기 구현을 위한 규칙을 자동으로 생성할 수 있고, 퍼지 추론을 위한 병렬성이 증가하였다. 규칙의 자동생성은 학습을 통한 소속 함수의 생성과 T-S 규칙의 결론부 함수 근사화 등을 통하여 가능하게 되었고, 퍼지화와 비퍼지화를 추론망에 포함시키고 조건부와 결론부를 병렬로 연결시킴으로 빠른 추론이 가능하다.

제안된 추론망은 퍼지 시스템의 설계자가 정하여야 할 파라미터들을 많이 감소 시켰다. 하지만 규칙의 수와 소속 함수의 퍼지 구간의 수를 결정하는 조건부 신경망의 경쟁층 뉴런의 수는 아직 설계자에 달려 있다. 기존의 제어기에서 이 부분은 설계자가 결정하여 왔지만, 자동화를 위해서는 입력 공간에서 타당한 군집의 수에 대한 연구가 선행되어야 할 것이다.

참고 문헌

1. Chuen Chien Lee, "Fuzzy Logic in Control Systems: Fuzzy Logic Controller - Part I-II," IEEE Transaction on System, Man, and Cybernetics, Vol. 20, No. 2, pp. 404-435, March/April, 1990.
2. Chin-Teng Lin, C. S. George Lee, "Neural-Network-Based Fuzzy Logic Control and Decision System," IEEE Trans. on Computers, Vol. 40, No. 12, pp. 1320-1336, December 1991.
3. Kaoru Hirota, Witold Pedrycz, "Fuzzy Modelling Environment for Designing Fuzzy Controllers," Fuzzy Sets and Systems, Vol. 70, pp. 287-301, 1995.
4. Sushmita Mitra and Sankar K. Pal, "Fuzzy Multi-Layer Perceptron, Inferencing and Rule Generation," IEEE Transaction on Neural Networks, Vol. 6, No. 1, pp 51-63, January, 1995.
5. Tomohiro Tagaki and Michio Sugeno, "Fuzzy Identification of Systems and Its Application to Modeling and Controls," IEEE Transaction on System, Man, and Cybernetics, Vol. SMC-15, No. 1, pp. 116-132, Januaray/February 1985.
6. James A. Freeman, David M. Skapura, Neural Networks Algorithms, Applications, and Programming Techniques, Addison Wesley, 1992.
7. 이상령, "퍼지 시스템에서의 신경망을 이용한 퍼지 규칙 생성과 추론망 구축," 석사학위논문, 서강대학교, 1994.
8. K. Tanaka and M. Sano, "Trajectory Stabilization of A Model Car via Fuzzy Control,"

Fuzzy Sets and Systems Vol. 70, pp. 155-170, 1995.

9. Li-Xin Wang, Jerry M. Mendel, "Generating Fuzzy Rules by Learning from Examples," IEEE Transaction on System, Man, and Cybernetics, Vol. 22, No. 6, pp 1414-1427, November/December, 1992.

10. Seong-Gon Kong and Bart Ksko, "Comparison of Fuzzy and Neural Truck Backer-Upper Control Systems," Proc, IJCNN-90, Vol. 3, pp.349-358, June, 1990.