

論文93-30B-10-9

## 퍼지 관계방정식의 해법을 위한 신경회로망 모델과 학습 방법

## (A Neural Network Model and Its Learning Algorithm for Solving Fuzzy Relational Equations)

全命根\*, 卞憎男\*\*

(Myung Geun Chun and Zeungnam Bien)

## 要約

본 논문에서는 볼록 결합된 퍼지 관계방정식을 푸는 방법을 제안한다. 이를 위해 퍼지 관계식의 구조를 갖는 신경회로망을 제안하고, 제안된 신경회로망에 대한 학습 알고리즘을 오류역전파의 개념을 이용하여 제안한다. 논문에서 다룬 퍼지 관계방정식은 일반적인 형태를 갖고 있으므로 보통의 퍼지 max-min 또는 min-max 관계방정식은 제안된 방법의 특수한 경우 보고 그 해를 쉽게 구할 수 있다. 더우기, 제시된 알고리즘이 신경회로망에 근기한 병렬 처리가 가능한 구조임을 감안하면 퍼지 제어, 지식 기반 시스템, 영상 인식시스템에서와 같은 부분의 실시간적용에 유용하게 사용될 수 있다.

## Abstract

In this paper, we present a method to solve a convexly combined fuzzy relational equation with generalized connectives. For this, we propose a neural network whose structure represents the fuzzy relational equation. Then we derive a learning algorithm by using the concept of back-propagation learning. Since the proposed method can be used for a general form of fuzzy relational equations, such fuzzy max-min or min-max relational equations can be treated as its special cases. Moreover, the relational structure adopted in the proposed neurocomputational approach can work in a highly parallel manner so that real-time applications of fuzzy sets are possible as in fuzzy logic controllers, knowledge-based systems, and pattern recognition systems.

## 1. 서론

Zadeh 교수에 의해 처음으로 제창된 퍼지 이론은

\*正會員, 三星電子, 生産技術 Center  
(SAMSUNG Electronics, Production  
Engineering Center)

\*\*正會員, 韓國科學技術院 電氣 및 電子工學科  
(Dept. of Electrical Eng. KAIST)

接受日字: 1992年 11月 28日

기존의 수학적 방법으로는 잘 정의될 수 없는 시스템의 모델링 및 분석에 널리 유용하게 응용되어 왔다. 이러한 퍼지이론의 한 분야로서, 퍼지 관계 방정식에 관한 이론 및 응용에 관한 연구는 Sanchez<sup>[12]</sup>에 의해 시작된 이후 또 다른 중요한 연구 대상이 되어 왔다. 퍼지 관계방정식의 중요성은, 인간의 지식을 사실과 규칙이 모여 각각이 서로관계를 갖으며, 그들의 관계가 가능성 분포를 갖는 것으로 볼 수 있는데 기인한다.<sup>[3]</sup> 즉, 인간의 지식을 하나의 대규모 퍼지 관계 시스템으로 볼 수 있기 때문이다. 또한 퍼지 관계

방정식에 기초한 퍼지 모델링은 비선형이며 시변 특성을 갖는 실제의 복잡한 시스템을 모델하고 제어하는데도 유용하게 응용되었다. [14] 이러한 관점에서 퍼지 관계식의 해를 구하는 것은 퍼지 시스템을 분석하고 설계하는데 있어서 중요함을 더해가고 있다.

이러한 퍼지 관계방정식의 해를 구하는 방법으로 여러가지 해석적인 방법들이 제안 되었다. [3] [4] [7] [8] [12] 그러나 이들 해석적인 방법의 대부분은 주어진 퍼지 입·출력 데이터에 대하여 그들을 모두 정확히 만족하는 퍼지 관계 행렬이 존재한다는 가정에 기초하고 있다. 그렇지만 이러한 가정은 실제의 상황에서는 만족되기가 쉽지 않다. 왜냐하면, 주어진 퍼지 입·출력 데이터들 사이에 정확한 일관성이 있는 경우는 드물기 때문이다.

퍼지 관계방정식의 해를 구하는 또 다른 방법으로 근사적인 해를 구하는 수치해석적인 방법이 제안되었다. [6] [11] Pedricz [6] 는 주어진 한 쌍의 퍼지 입·출력 데이터에 대하여 Newton의 방법에 의한 수치해석 방법을 제안하였다. 그렇지만 이 경우에 여러쌍의 퍼지 입·출력 데이터가 주어진 경우에 대해서 쉽게 확장되지 않는 단점이 있다. Saito와 Mukaidono [11] 는 MAX-MIN network이라는 단층 신경회로망의 일종을 제안하고 이것을 이용한 퍼지 관계 방정식의 해를 구하는 문제를 다루었다. 보통의 신경회로망이 각 뉴런에서 입력신호의 가중치 합(weighted sum)을구하는 반면 그들이 제안한 신경회로망은 각 뉴런에서 max-min 합성(max-min composition)을 수행한다. 그렇지만 저자들이 지적하듯이 제안한 학습방법은 정확한 해가 존재 할 경우에만 정상적으로 동작하는 단점이 있다. 최근, Pedricz [9] 는 [11] 의 신경회로망과 유사하게 동작하는 신경회로망을 제안하고주어진 퍼지 입·출력 데이터에 대하여 주어진 성능지수를 최적화하는 학습 방법을 제안하였다. 그러나 제안된 신경회로망은 은닉층을 포함하지 않는 단층 구조로 되어 있어 그 동작이 제한적이므로, max-min 관계방정식의 해를 구하기 위한 일반적인 학습방법을 제안하지 못하였다. 더우기 이러한 구조적 제약을 극복하기 위해서 퍼지 관계 방정식과는 무관한 bias항을 추가하여 학습 방법을 제시하고 있으므로 얻어진 해가 0이 아닌 bias를 포함하는 경우에 퍼지 관계식의 해가 되지 않는 문제가 있다.

본 논문에서는, 일반화된 max-min 합성과 min-max 합성을 수행하는 다층 신경회로망을 제안하고, 퍼지 관계방정식을 풀기 위한 학습방법을 오류 역전파의 개념을 이용하 제안하고자 한다. 제안된 방법은 일반적인 형태의 퍼지 관계방정식에 대하여 기술하고

있으므로 보통의 max-min 또는 min-max 퍼지 관계 방정식은 제안된 방법의 특수한 경우로 볼 수 있고 따라서 그 해를 쉽게 구할 수 있다.

논문의 구성은 다음과 같다. 2장에서는 max-min 퍼지 관계 방정식에 관련된 성질들을 기술하고 불록 결합된 퍼지 관계방정식에 대하여 설명한다. 3장에서는 퍼지 관계방정식을 풀기 위한 다층 신경회로망을 제안하고 그것의 학습방법을 제안한다. 4장에서 몇개의 수치예를 보인후 결론을 제시한다.

## II 불록 결합된 퍼지 관계방정식

이번 장에서는 먼저 퍼지 min-max 관계방정식을 해석적으로 푸는 방법에 대한 알려진 성질들을 알아본다.

$X$ 와  $Y$ 를 각각  $n$ -차원과  $m$ -차원의 공간이라 할 때,  $F(X) = x: X \rightarrow [0, 1]$  및  $F(Y) = y: Y \rightarrow [0, 1]$  를  $X$ 와  $Y$ 에서의 모든 퍼지 부분집합으로 이루어진 집합이라하고,  $R \in F(X \times Y)$ 을 퍼지 관계 행렬이라 하자.  $x = [x_1, x_2, \dots, x_n] \in F(X)$ 와  $y = [y_1, y_2, \dots, y_m] \in F(Y)$ 에 대하여 다음과 같은 퍼지 max-min 관계방정식을 생각해 보자.

$$x \circ R = y. \quad (1)$$

즉,

$$y_j = \bigvee_{1 \leq i \leq n} (x_i \wedge r_{ij}), 1 \leq j \leq m. \quad (2)$$

여기서,  $\wedge$ 와  $\bigvee$ 는 각각 min과 max를 나타낸다.

주어진 한쌍의  $x$ 와  $y$ 에 대하여,  $R$ 을  $x \circ R = y$ 의 모든 해의 집합이라 할때, 즉

$$\mathfrak{R} = \{R | x \circ R = y\}.$$

(1)의 해를 해석적으로 구하는 방법은 다음과 같이 주어진다.

정리 1 [12]: 만약  $R \neq 0$ 이면, (1)을 만족하는 해 중 가장 큰 퍼지 관계행렬  $\bar{R} = (\bar{r}_{ij}) \in \mathfrak{R}$  는  $x$ 와  $y$ 의 @ 합성에 의해 주어진다. 즉,  $R = x @ y$ ,  $n \times m$  행렬  $\bar{R} = [\bar{r}_{ij}] (i=1, 2, \dots, n, j=1, 2, \dots, m)$  와 같이 표기 할 때

$$\bar{r}_{ij} = \begin{cases} 1 & \text{if } x_i \leq y_j \\ y_j & \text{if } x_i > y_j \end{cases} \quad (3)$$

정리 2 [8]: 만약 퍼지 입력 출력 데이터  $(x^l, y^l)$  ( $l=1, 2, \dots, P$ )가 다음의 세가지 조건 즉, (1)  $x^l \circ R$

=  $y^i$ , (2) 만약  $i \neq i'$ 이면,  $x^i=1$ , 그렇지 않으면  $x^i=0$   
 (3)  $P=n$ 을 만족하면

$$\bigcap_{i=1}^P \bar{R}_i = R \quad (4)$$

여기서  $\bar{R}_i = x^i \alpha y^i$ .

그러나, 주어진 퍼지 데이터  $(x^i, y^i)$ 는 일반적으로 정리 2에 주어진 조건들을 모두 만족할 수 없다. 그러므로  $(x^i, y^i)$  ( $i=1, 2, \dots, P$ )에 대하여  $y^i = x^i \circ R$ 를 만족하는  $R$ 을 해석적으로 구하는 방법은 쉽지 않다.

일반적으로 퍼지 관계 방정식은 위와 같은 max-min 또는 min-max 합성에 기반을 두고 있다. 그렇지만 Ohsato<sup>[5]</sup>가 지적했듯이 좀더 복잡한 퍼지 시스템에 있어서는 좀더 복잡한 형태의 퍼지 관계방정식을 필요로 한다. 이런 점에서, Ohsato와 Sekiguchi<sup>[5]</sup>는 max-min 합성과 min-max 합성  $\Delta$ 에 기반을 둔다음과 같은 블록 결합된 퍼지 관계식을 제안하였다.

$$\lambda \circ (x \circ \bar{R}) + \bar{\lambda} \circ (x \Delta \bar{R}) = y \quad (5)$$

여기서  $x \in F(X)$ 와  $y \in F(Y)$ 는 주어진 퍼지집합이며  $\lambda \in F(Y)$  ( $\bar{\lambda} = [1 \dots 1]^T - [\lambda_1, \lambda_2, \dots, \lambda_m]^T$ )는 임의의 퍼지 결합자 (fuzzy convex combinator),  $\hat{R}, \check{R} \in F(X \times Y)$ 는 알려지지 않은 두개의 퍼지 관계행렬이다. 그들은 한 쌍의 주어진  $x$ 와  $y$ 에 대하여 해가 존재 할 수 있는  $\lambda$ 의 범위를 찾는 방법과 하나의 정해진  $\lambda$ 에 대하여 (5)를 만족하는 가장 큰  $R$ 와  $\check{R}$ 를 구하는 방법을 제시하였다. 그렇지만, 여러쌍의  $x$ 와  $y$ 가 동시에 주어지는 경우에 대해서는 고려하지 않았다. 이 경우 다른 해석적인 방법들과 마찬가지로 해를 구할 수 없는 경우가 많다.

본 논문에서 다루고자 하는 퍼지 관계 방정식은 주어진  $\lambda$ 에 대하여 ( $\lambda \in \text{convex}$ )보다 좀더 일반적인 다음과 같은 형태이다.

$$\lambda \circ (x \circ_p \hat{R}) + \bar{\lambda} \circ (x \Delta_p \check{R}) = y \quad (6)$$

즉,

$$y_j = \lambda_j \max \left( \min_p \left( x_i, \check{r}_{ij} \right) \right) + (1 - \lambda_j) \min \left( \max_p \left( x_i, \hat{r}_{ij} \right) \right), 1 \leq j \leq m. \quad (7)$$

여기서  $\circ_p$ 와  $\Delta_p$ 는 각각 일반화된 max-min 합성

과 min-max 합성을 나타내며  $\min_p$ 과  $\max_p$ 는 다음과 같이 정의 된다.

$$\min_p(x, y) = 1 - \min \left[ 1, \left\{ (1-x)^p + (1-y)^p \right\}^{1/p} \right] \text{ for } p \geq 1 \quad (8)$$

$$\max_p = \min \left[ 1, \left( x^p + y^p \right)^{1/p} \right] \text{ for } p \geq 1 \quad (9)$$

위의 일반화된 min,  $\min_p$ 와 일반화된 max,  $\max_p$ 는 Yager<sup>[13]</sup>에 의해서 처음으로 제안되었다.

위의 기술을 종합하여, 본 논문에서 다루고자 하는 문제를 정리하면 다음과 같다.

[문제] 주어진  $\lambda \in F(Y)$ 에 대하여 퍼지 데이터  $(x^i, y^i)$  ( $i=1, 2, \dots, P$ )이 주어지고  $y^i$ 에 대한 근사값을  $\bar{y}^i$ 이라 할때, 각각의 퍼지 데이터에 대한 성능지수  $E_i = \frac{1}{2} \sum_{j=1}^m (y_j^i - \bar{y}_j^i)^2$ 의 합을 최소화 하는  $R$ 과  $\check{R}$ 을 구한다. 즉,

$$\min_{\hat{R}, \check{R} \in F(X \times Y)} \sum_{i=1}^P \sum_{j=1}^m \left[ y_j^i - \left\{ \lambda_j \vee \left( x_i^j, \check{r}_{ij} \right) + (1 - \lambda_j) \wedge \left( x_i^j, \hat{r}_{ij} \right) + (1 - \lambda_j) \right\} \right]^2 \quad (10)$$

위와 같은 문제에서 전역최적해(global optimal solution)를 찾는 것은 실제로 쉽지 않다. 여기서는 보통의 신경회로망에서 많이 쓰이는 steepest descent 알고리즘에 의해 해를 찾는 방법을 제시하고자 한다. 그러므로 구해진 해가 지역최적해(local optimal solution)일 수가 있다.

Remark 1: 만약  $p = \infty$ 이면,  $\min_p$ 와  $\max_p$ 는 보통의 min, max가 된다. 따라서  $p = \infty$ 이면 (6)는 (5)가 된다.

Remark 2: 만약  $\lambda = [1 \ 1 \ \dots \ 1]^T$ 이고  $p = \infty$ 이면, (6)는 보통의 퍼지 max-min 관계 방정식  $y = x \circ R$ 가 된다.

Remark 3: 만약  $\lambda = [1 \ 1 \ \dots \ 1]^T$ 이면, (6)는  $y = x \circ_p R$ 이 된다. 이런 형태의 퍼지 관계방정식에 대한 해석적인 방법은 [7]에 기술되어 있다. 그렇지만 제시된 방법은 주어진 모든 퍼지 집·출력 데이터가  $y^i = x^i \circ_p R$ 을 만족하지 않으면 해를 구할 수 없다.

### III 퍼지 관계방정식과 신경회로망

퍼지논리와 신경회로망의 융합에 관한 연구는 최근에 활발히 진행되고 있다.<sup>[11][12]</sup> 그러나, 보통의 신경회로망이 퍼지시스템에 대한 학습 메카니즘에 사용되었을 때, 성공적인 학습 후에 얻어진 최종 가중값들은 대상 퍼지시스템의 구조에 대한 정보를 제공하지

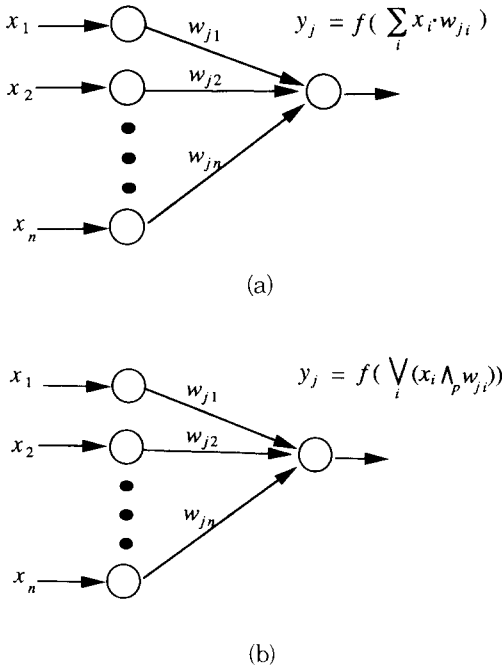


그림 1. 보통의 신경회로망과 max-min<sub>p</sub> 신경회로망  
 (a) 보통의 신경회로망  
 (b) max-min<sub>p</sub> 신경회로망  
 Fig. 1. A usual neural network and max-min<sub>p</sub> neural network.  
 (a) A usual neural network,  
 (b) A max-min<sub>p</sub> neural network.

않고 입·출력 mapping 관계만을 모사하게 된다. 이는, 퍼지 시스템이 min과 max에 의해 동작되는 반면 보통의 신경회로망은 곱셈과 덧셈으로 이루어지는 서로 다른 구조적 특성을 갖고 있기 때문이다. Saito와 Mukaidono<sup>[11]</sup> 그리고 Pedricz<sup>[9]</sup>는 신경회로망과 max-min 퍼지 관계방정식과의 유사성으로 부터 min, max에 의해 동작되는 단층 신경회로망을 제안하였다. 그렇지만 그들이 제안한 신경회로망은 은닉층을 포함하지 않는 구조적 제약을 갖고 있다.

그림 1(a)에서 보듯이 보통의 신경회로망은 각각의 뉴런에서 가중치 합을 구하며 각각의 가중치들은 학습규칙에 의해서 성능지수를 최적화 하는 쪽으로 변화되게 된다. 반면, 그림 1(b)에서와 같은 본 논문에서 사용되는 신경회로망에서는 각 뉴런에서 max-min<sub>p</sub> 합성을 수행하게 되는데, 이때 각 뉴런간의 가중값들은 관계 행렬을 나타내게 된다. (6)와 같이 주

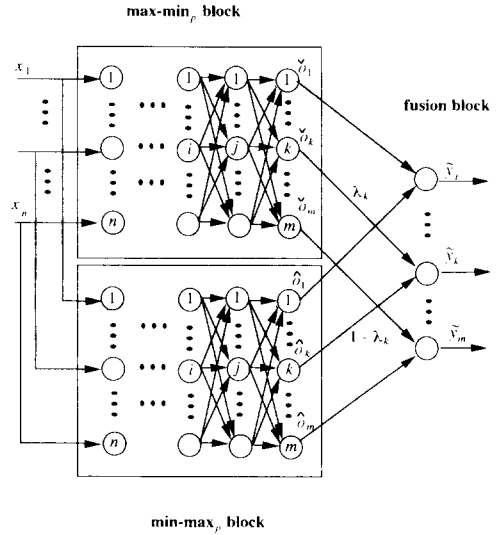


그림 2. 제안된 신경회로망 구조  
 Fig. 2. Proposed neural network scheme.

어지는 퍼지 관계 방정식에 대하여, 그림 2와 같은 신경회로망 구조를 제안한다. 제안된 신경 회로망의 크 계 3개의 block으로 나눌 수 있는데, 하나는 max-min<sub>p</sub> block이고 또 하나는 min-max<sub>p</sub> block 그리고 fusion block이다. 여기서 max-min<sub>p</sub>, min-max<sub>p</sub> block은 은닉층을 갖을 수 있다. 즉,  $I^{n \times m}$ 을  $[0, 1]^n \times [0, 1]^m$ 에서의  $(n \times m)$ -차원 공간이라 하자. 그러면 max-min<sub>p</sub> block  $x \circ_p R$ 에서 관계 행렬  $R \in I^{n \times m}$ 이라 하면,

$$\check{R} = \check{R}^1 \circ_p \check{R}^2 \circ_p \check{R}^3 \dots \circ_p \check{R}^q \text{ for } q \geq 1 \tag{11}$$

의 형태로 나누어 생각 할 수 있다. 여기서  $\check{R}^1 \in I^{n \times h_1}, \check{R}^2 \in I^{h_1 \times h_2}, \dots, \check{R}^q \in I^{h_{q-1} \times m}$ . 의 (11)관계로 부터, 우리는  $R$ 을 q개의 층으로 이루어진 신경회로망으로 부터 좀더 일반적인 방법으로 구할 수 있다. 즉, 신경회로망의  $i(1 \leq i \leq q)$ 번째 층은  $\check{R}^i$ 을 가중값으로 하는 것으로 생각할 수 있으므로, 학습규칙에 의해 각각의  $\check{R}^i$ 을 구하여 (11)의 관계로 부터 원하는  $R$ 을 구할 수 있다.

그러면, 그림 2와 같이 주어진 신경회로망의 동작을 살펴보고 오류 역전파 이론에 기초한<sup>[10]</sup> 학습 규칙을 유도하여 보자. 앞으로, 기호를 간략히 하기위해 (10)에서 첨자  $l$ 을 생략하고자 한다.  $\hat{o}$ 과  $\check{o}$ 로 min max<sub>p</sub>과 max min<sub>p</sub> block의 출력을 나타내면, fusion block의 뉴런은 다음의 동작을 한다.

$$\check{y}_k = \lambda_k \check{o}_k + (1 - \lambda_k) \hat{o}_k \quad (k=1, 2, \dots, m) \quad (12)$$

다음으로, max-min<sub>p</sub> block의 동작을 살펴 보자.

$$\check{net}_k = \max_j \min_p (\check{r}_{kj}, \check{o}_j) \quad (13)$$

그리고

$$\check{o}_k = f(\check{net}_k)$$

여기서 출력함수 f는 다음과 같이 선택하였다.

$$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ x & \text{for } 0 \leq x \leq 1 \\ 1 & \text{for } x > 1 \end{cases} \quad (14)$$

본 논문에서 출력함수 f의 미분값은 항상 1로 주어진다. 왜냐하면 f에 주어지는 값 x가 0 ≤ x ≤ 1이기 때문이다.

가중치의 갱신은  $-\partial E / \partial \check{r}_{kj}$ 에 비례하는 변화량  $\Delta \check{r}_{kj}$ 에 의해 이루어진다. 즉,

$$\Delta \check{r}_{kj} = -\eta \frac{\partial E}{\partial \check{r}_{kj}} \quad (15)$$

여기서  $\eta \in [0, 1]$ 는 학습계수이다<sup>110</sup>.

그리고  $\frac{\partial E}{\partial \check{r}_{kj}} = \frac{\partial E}{\partial \check{net}_k} \frac{\partial \check{net}_k}{\partial \check{r}_{kj}}$ . 그러면

$$\frac{\partial \check{net}_k}{\partial \check{r}_{kj}} = \begin{cases} \left( (1 - \check{r}_{kj})^{p-1} \cdot (1 - \check{r}_{kj})^p + (1 - \check{o}_j)^{p-1} \right)^{-1} \\ \text{if } (\check{r}_{kj} \wedge_p \check{o}_j) \geq \vee_{i \neq k} (\check{r}_{ij} \wedge_p \check{o}_i) \text{ and } \left( (1 - \check{r}_{kj})^p + (1 - \check{o}_j)^p \right)^{1/p} \leq 1 \\ 0 & \text{otherwise} \end{cases}$$

$$\check{s}_{ki} = \frac{\partial \check{net}_k}{\partial \check{r}_{ij}} \text{ 라 두고 } \check{\delta}_i = \frac{\partial E}{\partial \check{net}_i} \text{ 라 놓으면}$$

$$\check{\delta}_i = \frac{\partial E}{\partial \check{o}_i} \frac{\partial \check{o}_i}{\partial \check{net}_i}$$

여기서,

$$\frac{\partial E}{\partial \check{o}_i} = -\lambda_i \cdot (y_i - \lambda_i \cdot \check{o}_i - (1 - \lambda_i) \cdot \hat{o}_i)$$

그리고

$$\frac{\partial \check{o}_i}{\partial \check{net}_i} = 1.$$

그러면,

$$\check{\delta}_i = \lambda_i \cdot (y_i - \lambda_i \cdot \check{o}_i - (1 - \lambda_i) \cdot \hat{o}_i)$$

t를 학습단계 (learning step)로 놓으면, 출력층에 대한 가중치 갱신 수식은 다음과 같다.

$$(16)$$

여기서

$$\check{r}_{ij}(t+1) = \check{r}_{ij}(t) + \Delta \check{r}_{ij} \quad (17)$$

이제, max-min<sub>p</sub> block의 은닉층에 대한 학습식을 유도하여 보자.

$$\Delta \check{\delta}_{ij} = \eta \check{\delta}_{ij} \check{s}_{ij} = \eta \lambda_k (y_k - \lambda_k \cdot \check{o}_k - (1 - \lambda_k) \cdot \hat{o}_k) \check{s}_{ij}$$

여기서  $\check{r}_{ij}^h$ 은 max-min<sub>p</sub> block의 i번째 층의 뉴런으로 부터 그다음 j번째 층의 뉴런으로의 가중치이며 첨자 "h"는 그것이 은닉층에 속함을 나타낸다. 그러면,

$$\check{o}_j = f(\check{net}_j^h)$$

여기서 출력함수 f는 (14)와 같다.

$$\check{s}_{ij}^h = \frac{\partial \check{net}_j^h}{\partial \check{r}_{ij}^h} \text{ 이라 놓으면,}$$

$$\check{s}_{ij}^h = \begin{cases} \left( (1 - \check{r}_{ij}^h)^{p-1} \cdot \left( (1 - \check{r}_{ij}^h)^p + (1 - \check{o}_j)^p \right)^{1/p} \right)^{-1} \\ \text{if } (\check{r}_{ij}^h \wedge_p \check{o}_j) \geq \vee_{i \neq j} (\check{r}_{ij}^h \wedge_p \check{o}_i) \text{ and } \left( (1 - \check{r}_{ij}^h)^p + (1 - \check{o}_j)^p \right)^{1/p} \leq 1 \\ 0 & \text{otherwise} \end{cases}$$

그리고

$$\Delta \check{r}_{ij}^h = -\eta \frac{\partial E}{\partial \check{r}_{ij}^h} \quad (18)$$

$$= -\eta \frac{\partial E}{\partial \check{net}_j^h} \frac{\partial \check{net}_j^h}{\partial \check{r}_{ij}^h} \quad (19)$$

$$= \eta \left( -\frac{\partial E}{\partial \check{net}_j^h} \right) \check{s}_{ij}^h \quad (20)$$

또한  $-\partial E / \partial \check{net}_j^h$  이라 놓고 chain rule을 적용하면

$$\check{\delta}_j^h = \frac{\partial E}{\partial \check{o}_j} \frac{\partial \check{o}_j}{\partial \check{net}_j^h} = \frac{-\partial E}{\partial \check{o}_j} = -\sum_i \frac{\partial E}{\partial \check{net}_i} \frac{\partial \check{net}_i}{\partial \check{o}_j} \quad (21)$$

$$= \sum_i \left( -\frac{\partial E}{\partial \check{net}_i} \right) \frac{\partial}{\partial \check{o}_j} \left( \max \min_p (\check{r}_{ij}, \check{o}_i) \right) \quad (22)$$

$$= \sum_i \left( -\frac{\partial E}{\partial \check{net}_i} \right) w_{ij} = \sum_i \check{\delta}_i w_{ij} \quad (23)$$

여기서

$$\check{w}_{kj} = \begin{cases} \left( (1-\check{o}_j)^{p-1} \left( (1-\check{r}_{kj})^p + (1-\check{o}_j)^p \right)^{\frac{1}{p}-1} \right) \\ \text{if } (\check{r}_{kj} \wedge_p \check{o}_j) \geq \vee_{s,r_j} (\check{r}_{ks} \wedge_p \check{o}_s) \text{ and } \left( (1-\check{r}_{kj})^p + (1-\check{o}_j)^p \right)^{\frac{1}{p}} \leq 1 \\ 0 \quad \text{otherwise} \end{cases}$$

$$\check{w}_{kj} = \begin{cases} \left( (\check{o}_j)^{p-1} \left( (\check{r}_{kj})^p + (\check{o}_j)^p \right)^{\frac{1}{p}-1} \right) \\ \text{if } (\check{r}_{kj} \vee_p \check{o}_j) \leq \wedge_{s,r_j} (\check{r}_{ks} \vee_p \check{o}_s) \text{ and } \left( (\check{r}_{kj})^p + (\check{o}_j)^p \right)^{\frac{1}{p}} \leq 1 \\ 0 \quad \text{otherwise} \end{cases}$$

위의 수식들로 부터, max-min<sub>p</sub> block의 은닉층에 대한 가중치 갱신 수식은 다음과 같이 주어진다.

$$\check{r}_{ji}^h(t+1) = \check{r}_{ji}^h(t) + \Delta \check{r}_{ji}^h \quad (24)$$

여기서

$$\Delta \check{r}_{ji}^h = \eta \check{\delta}_i^h \check{s}_{ji}^h = \eta \check{s}_{ji}^h \sum_k k \eta \check{\delta}_k \check{w}_{jk}^h \quad (25)$$

min-max<sub>p</sub>에 대해서는 max-min<sub>p</sub> block에 대한 것들과 유사하게 다음과 같이 유도된다. 출력층에 대해서는

$$\hat{r}_{ij}(t+1) = \hat{r}_{ij}(t) + \Delta \hat{r}_{ij} \quad (26)$$

여기서

$$\Delta \hat{r}_{ij} = \eta \hat{\delta}_i \hat{s}_{ij} = \eta (1 - \lambda_k) \left( \lambda_k - \lambda_k \cdot \check{o}_k - (1 - \lambda_k) \cdot \check{o}_k \right) \hat{s}_{ij} \quad (27)$$

그리고

$$\hat{s}_{ij} = \begin{cases} (\hat{r}_{ij})^{p-1} \cdot \left( (\hat{r}_{ij})^p + (\hat{o}_j)^p \right)^{\frac{1}{p}-1} \\ \text{if } (\hat{r}_{ij} \vee_p \hat{o}_j) \leq \wedge_{s,r_j} (\hat{r}_{is} \vee_p \hat{o}_s) \text{ and } \left( (\hat{r}_{ij})^p + (\hat{o}_j)^p \right)^{\frac{1}{p}} \leq 1 \\ 0 \quad \text{otherwise} \end{cases} \quad (28)$$

은닉층에 대해서는

$$\check{r}_{ji}^h(t+1) = \check{r}_{ji}^h(t) + \Delta \check{r}_{ji}^h \quad (29)$$

여기서

$$\Delta \check{r}_{ji}^h = \eta \hat{\delta}_i^h \check{s}_{ji}^h = \eta \check{s}_{ji}^h \sum_k \hat{\delta}_k \check{w}_{jk}^h$$

그리고

$$\check{s}_{ij} \frac{\partial \hat{r}_{ij}^h}{\partial \check{r}_{ji}^h} = \begin{cases} (\check{r}_{ji}^h)^{p-1} \cdot \left( (\check{r}_{ji}^h)^p + (\check{o}_j)^p \right)^{\frac{1}{p}-1} \\ \text{if } (\check{r}_{ji}^h \vee_p \check{o}_j) \leq \wedge_{s,r_j} (\check{r}_{is}^h \vee_p \check{o}_s) \text{ and } \left( (\check{r}_{ji}^h)^p + (\check{o}_j)^p \right)^{\frac{1}{p}} \leq 1 \\ 0 \quad \text{otherwise} \end{cases}$$

제안된 학습 알고리즘의 흐름은 보통의 오류 역전파 알고리즘의 흐름과 유사한데, 다음과 같이 간략히 정리 될 수 있다.

[단계 1] max-min<sub>p</sub>와 min-max<sub>p</sub> block의 각 층에 대한 연결행렬값을 임의의 값을 주므로서  $\check{R}$ 과  $\hat{R}$ 에 대한 초기값  $\check{R}(0)$ 과  $\hat{R}(0)$ 을 부여한다.

[단계 2] 각층의  $\check{R}(t)$ 와  $\hat{R}(t)$  ( $t=1, 2, \dots$ )을 각각, 주어진  $(x^i, y^i)$  ( $1 \leq i \leq P$ ),  $p, \lambda$ 에 대하여 (16) (24)과 (26) (29)을 이용하여 알고리즘의 종료 조건으로 정의된 기준이 되는 성능 지수이하가 될 때까지 갱신해 나간다. 만약 원치않는 local minimum에 수렴하는 경우이면 단계 1로 되돌아 간다.

#### IV 수치 예

다음과 같이 주어진 퍼지 임·출력 데이터에 대하여 제안된 알고리즘을 수행시켜 보자.

$$x^1 = [0.500 \ 0.800 \ 0.400], \quad y^1 = [0.7986 \ 0.5000 \ 0.5000]$$

$$x^1 = [0.300 \ 0.900 \ 0.600], \quad y^1 = [0.8000 \ 0.4000 \ 0.3021]$$

$$x^1 = [0.700 \ 0.300 \ 0.100], \quad y^1 = [0.3000 \ 0.6000 \ 0.4000]$$

먼저,  $\lambda = [1 \ 0.5 \ 0]^T$  그리고  $p=100$ 이 경우에 대하여 하나의 은닉층을 갖고 은닉층은 4개의 뉴런으로 구성된 신경회로망에 대하여,  $\eta=0.2$ 으로 하고 제한된 알고리즘을 수행하였다.  $\check{R}$ 와  $\hat{R}$ 에 대한 초기값은 max-min<sub>p</sub> block에 대하여 출력층을  $\check{R}^o(0)$ , 은닉층을  $\check{R}^h(0)$ 로 나타내었을때 다음과 같이 임의의 값으로 초기화 하였다.

$$\check{R}^o(0) = \begin{bmatrix} 0.5350 & 0.3665 & 0.6595 \\ 0.9392 & 0.0917 & 0.4468 \\ 0.3061 & 0.9605 & 0.5837 \\ 0.2915 & 0.4453 & 0.9484 \end{bmatrix}$$

그리고

$$\check{R}^h(0) = \begin{bmatrix} 0.5170 & 0.6877 & 0.0444 & 0.9589 \\ 0.5046 & 0.5385 & 0.8038 & 0.5803 \\ 0.3719 & 0.4196 & 0.1004 & 0.6850 \end{bmatrix}$$

min-max<sub>p</sub> block에 대하여 출력층을  $\hat{R}^o(0)$ , 은닉

층을  $\hat{R}^h(0)$ 로 나타내었을때 다음과 같이 임의의 값으로 초기화 하였다.

$$\hat{R}^o(0) = \begin{bmatrix} 0.6417 & 0.6342 & 0.7438 \\ 0.1087 & 0.5237 & 0.6530 \\ 0.8162 & 0.1679 & 0.6700 \\ 0.0604 & 0.3744 & 0.1152 \end{bmatrix}$$

그리고

$$\hat{R}^h(0) = \begin{bmatrix} 0.7397 & 0.2140 & 0.3629 & 0.8140 \\ 0.5593 & 0.0611 & 0.9681 & 0.2646 \\ 0.9315 & 0.5791 & 0.1758 & 0.9039 \end{bmatrix}$$

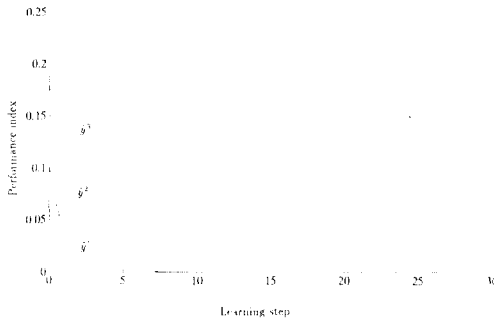


그림 3.  $p=100$ ,  $\lambda = [1 \ 0.5 \ 0]^T$ ,  $\eta = 0.2$ 일때의 학습과정

Fig. 3. Learning processes with  $p=100$ ,  $\lambda = [1 \ 0.5 \ 0]^T$ , and  $\eta = 0.2$ .

그림 3에 나타난 바와 같은 학습과정을 거쳐서 다음과 같은 최종 관계 행렬을 구할 수 있었다.

$$\check{R} = R^h \circ_p \check{R}^o = \begin{bmatrix} 0.3003 & 0.7116 & 0.9484 \\ 0.8000 & 0.5691 & 0.5691 \\ 0.4196 & 0.6850 & 0.6850 \end{bmatrix}$$

$$\hat{R} = \hat{R}^h \Delta_p \hat{R}^o = \begin{bmatrix} 0.2140 & 0.2860 & 0.3002 \\ 0.1087 & 0.4557 & 0.2647 \\ 0.5791 & 0.4521 & 0.5791 \end{bmatrix}$$

위의 구해진 관계 행렬에 의하면  $\bar{y}^1 = [0.7986 \ 0.5106 \ 0.5000]$ ,  $\bar{y}^2 = [0.8000 \ 0.4500 \ 0.3022]$ ,  $\bar{y}^3 = [0.3003 \ \sim 0.5760 \ \sim 0.3000]$  이 되고 이들에 대한 성능 지수는  $E_1 = 0.0001$ ,  $E_2 = 0.0025$ ,  $E_3 = 0.0106$ 와 같다.

위와 동일한 퍼지 데이터와  $\lambda$ 에 대하여 동일한 신경회로망 구조를 가지고  $p=1$ 인 경우에 대하여  $\eta = 0.2$ 를 가지고 학습 알고리즘을 적용시켰다. 이때의 초기

가중치는 다음과 같이 임의의 값으로 부여되었다.

$$\check{R}^o(0) = \begin{bmatrix} 0.4679 & 0.5717 & 0.4985 \\ 0.2872 & 0.8024 & 0.9554 \\ 0.1783 & 0.0331 & 0.7483 \\ 0.1537 & 0.5344 & 0.5546 \end{bmatrix}$$

$$\check{R}^h(0) = \begin{bmatrix} 0.8907 & 0.1598 & 0.1304 & 0.0030 \\ 0.6248 & 0.2128 & 0.0910 & 0.4143 \\ 0.8420 & 0.7147 & 0.2746 & 0.0269 \end{bmatrix}$$

그리고

$$\hat{R}^o(0) = \begin{bmatrix} 0.7098 & 0.3175 & 0.6813 \\ 0.9379 & 0.8870 & 0.3858 \\ 0.2399 & 0.6521 & 0.3877 \\ 0.1809 & 0.1503 & 0.4997 \end{bmatrix}$$

$$\hat{R}^h(0) = \begin{bmatrix} 0.1475 & 0.5901 & 0.1482 & 0.1418 \\ 0.5872 & 0.9554 & 0.9833 & 0.5649 \\ 0.8456 & 0.5561 & 0.4088 & 0.2521 \end{bmatrix}$$

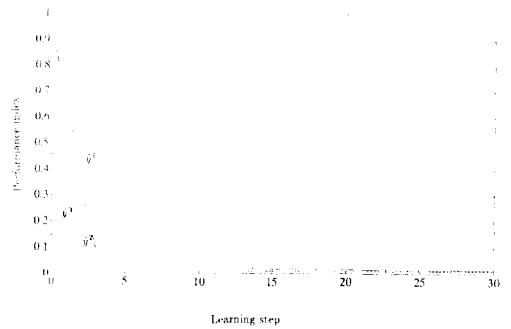


그림 4.  $p=1$ ,  $\lambda = [1 \ 0.5 \ 0]^T$ ,  $\eta = 0.2$ 일때의 학습과정

Fig. 4. Learning processes with  $p=1$ ,  $\lambda = [1 \ 0.5 \ 0]^T$ , and  $\eta = 0.2$ .

그림 4와 같은 학습을 거쳐서 다음과 같은 최종 가중치들을 구할 수 있었다.

$$\check{R} = R^h \circ_p \check{R}^o = \begin{bmatrix} 0.6590 & 0.3504 & 0.2491 \\ 0.5872 & 0.5843 & 0.4830 \\ 0.8456 & 0.5561 & 0.2521 \end{bmatrix}$$

그리고

$$\hat{R} = \hat{R}^h \Delta_p \hat{R}^o = \begin{bmatrix} 0.6590 & 0.3504 & 0.2491 \\ 0.8929 & 0.5843 & 0.4830 \\ 0.7504 & 0.5241 & 0.6736 \end{bmatrix}$$

구해진 관계 행렬에 의하면  $\bar{y}^1 = [0.6929 \ 0.5455 \ 0.5016]$ ,  $\bar{y}^2 = [0.7929 \ 0.4955 \ 0.3016]$ ,  $\bar{y}^3 = [0.3590 \ 0.4737 \ 0.3998]$  이 되고, 각각의 성능지수는  $E_1 = 0.0132$ ,  $E_2 = 0.0092$ ,  $E_3 = 0.0194$ 이 된다.

여러가지 경우에 대하여 simulation을 수행한 결과 제안된 학습 알고리즘 대하여 다음과 같은 정성적인 성질들을 얻을 수 있었다. 비교적 많은 뉴론을 갖는 은닉층을 늘릴수록 계산시간은 늘어나지만 학습단계는 줄어들었으며, 만족스럽지 않은 local minimum에 빠지는 것을 막을 수 있었다. 이때, 수렴속도를 증가시키기 위해서는, 예를 들어,  $W$ 의 가중치 갱신 수식에서 다음과 같은 형태로 momentum<sup>10)</sup>을 추가하면 수렴속도를 증가시킬 수 있다.

$$W(t+1) = W(t) + \Delta W(k) + \beta \Delta W(t-1) \text{ for } t \geq 1$$

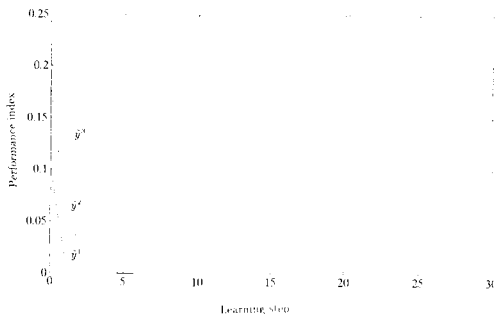


그림 5.  $p=100, \lambda = [1 \ 0.5 \ 0]^T, \eta = 0.2, \beta = 0.4$ 일때의 학습과정

Fig. 5. Learning processes with  $p=100, \lambda = [1 \ 0.5 \ 0]^T$ , and  $\eta = 0.2$ , and  $\beta = 0.4$ .

여기서  $\beta$ 는 momentum 계수이다. 그림 5는 그림 3에서 사용된 똑같은 parameter에 대하여,  $\beta = 0.4$ 인 momentum 항을 추가 했을 때의 학습을 나타낸다.

그림에서 알 수 있듯이 같은 결과를 좀더 빠른 수렴 속도로 얻을 수 있음을 알 수 있다.

### V 결론

본 논문에서는 퍼지 관계방정식을 풀기 위한 다층 신경회로망과 그것의 학습 방법을 제시하였다. 제시된 수치예들을 통하여 제안된 방법이 퍼지 관계 방정식의 근사해를 구하는데 유용하게 쓰일 수 있음을 보였다. 본 논문에서 대상으로 하는 퍼지 관계방정식이 일반화된 connective에 기초를 두고 있고, max-min과 min-max의 퍼지 관계식이 불록 결합된 형태이므로 보통의 min-max, max-min 퍼지 관계 방정식은 제시된 퍼지 관계방정식의 특수한 형태로 볼 수 있다. 더우기, 제시된 알고리즘이 신경회로망에

근거하기 때문에 병렬 처리가 가능한 구조를 감안하면 퍼지 제어, 지식 기반 시스템, 영상 인식시스템에서와 같은 부분에서의 실시간적용에 유용하게 사용되리라 기대된다. 앞으로 주어진 알고리즘에서 은닉층의 갯수 및 뉴론의 갯수와 알고리즘의 수렴성에 대한 관계의 정량적인 분석이 더 연구 되어야 한다고 생각된다.

### 參考文獻

- [ 1 ] J.M. Keller and H. Tahani, "Implementation of conjunctive and disjunctive fuzzy logic rules with neural networks," *Internat. J. Approximate Reasoning*, vol. 6 pp.221-240, 1992.
- [ 2 ] J.M. Keller, R.R. Yager, and H. Tahani, "Neural network implementation of fuzzy logic," *Fuzzy Sets and Systems*, vol. 45 1-12, 1992.
- [ 3 ] A. Di Nola, W. Pedricz, E. Sanchez and S. Sessa, *Fuzzy relational equations and their applications to knowledge engineering*, Kluwer Academic Publishers, Dordrecht, 1989.
- [ 4 ] A. Di Nola, W. Pedricz, S. Sessa, and E. Sanchez, "Fuzzy relational equations theory as a basis of fuzzy modelling: an overview," *Fuzzy Sets and Systems* vol. 40, pp. 415-429, 1991.
- [ 5 ] A. Ohsato and T. Sekiguchi, Convexly combined form of fuzzy relational equations and its applications to knowledge representation, *Proc. of IEEE International Conference on Systems, Man and Cybernetics*, vol. 1 Bombay and New Delhi, India, pp. 294-299, 1983.
- [ 6 ] W. Pedricz, "Numerical and applicational aspects of fuzzy relational equations," *Fuzzy Sets and Systems* vol. 11, pp. 1-18, 1983.
- [ 7 ] W. Pedricz, "Fuzzy relational equations with generalized connectives and their applications," *Fuzzy Sets and Systems* vol. 10, pp. 185-201, 1983.



- [8] W. Pedrycz, E. Czogala, K. Hirota, "Some remarks on the identification problem in fuzzy systems," *Fuzzy Sets and Systems* vol. 12, pp. 185-189, 1984.
- [9] W. Pedrycz, "Neurocomputations in relational systems," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 13, no. 3, pp. 289-297, 1991.
- [10] D.E. Rumelhart, J.L. McClelland, and the PDP Research Group, *Parallel distributed processing: Explorations in the microstructures of cognition*, Cambridge, M.I.T. Press, vol.1, 1986.
- [11] T. Saito and M. Mukaidono, A learning algorithm for MAX-MIN network and its application to solve fuzzy relation equations, *Proc. of IFSA '91*, Brussels, Belgium, vol. AI pp. 184-187.
- [12] E. Sanchez, "Resolution of compositional fuzzy relational equations," *Information and Control*, vol. 30, pp. 38-48, 1976.
- [13] R.R. Yager, "On a general class of fuzzy connectives", *Fuzzy Sets and Systems* vol. 4, pp. 235-242, 1980.
- [14] C. W. Xu and Y.-Z. Lu, "Fuzzy model identification and self-learning for dynamic systems," *IEEE Trans. on Systems Man Cybernetics*, vol. 17, no. 4, pp. 683-689, 1987.

---

 著者紹介
 

---

## 全命根(正會員)



1964年 7月 17日生. 1987年 2月 부산대학교 전자공학과 졸업(공학사). 1989年 2月 한국과학기술원 전기 및 전자공학과 졸업(공학석사). 1993年 8月 한국과학기술원 전기 및 전자공학과 졸업(공학박사). 1993年 8月 ~ 현재 삼성전자 생산기술 센터 자동화 연구소. 주관심분야는 Petri Net, Neural Net, Fuzzy Logic 등임.

## 卞增男(正會員) 第 29卷 B編 第 7號 參照

한국과학기술원 전기 및 전자공학과 교수