

## 퍼지 제어 기법을 이용한 FCM 기반 RBF 네트워크

김태형\*, 박충식\*\*, 김광백\*

### FCM-based RBF Network Using Fuzzy Control Method

Tae-Hyung Kim \*, Choong-Shik Park \*\*, Kwang-Baek Kim \*

#### 요약

FCM 기반 RBF 네트워크는 서로 다른 학습 구조가 결합된 혼합형 모델로서, 입력층과 중간층의 학습 구조는 FCM 알고리즘을 적용하고, 중간층과 출력층 사이의 학습 구조는 Max\_Min 신경망을 적용한다. 입력층과 중간층의 학습시 입력 벡터와 중간층의 노드 중에서 중심과 입력 벡터간의 가장 가까운 노드를 승자 노드로 선택하여 출력층으로 전달한다. 그리고 중간층과 출력층 사이의 학습 구조는 Max\_Min 신경망을 적용하여 중간층의 승자 뉴런이 출력층의 입력벡터로 적용한다. 하지만 많은 패턴이 입력벡터로 제시될 경우 학습 성능이 저하되는 단점이 있다. 따라서 본 논문에서는 중간층과 출력층의 학습 구조인 Max\_Min 알고리즘의 학습 성능을 개선시키기 위해 퍼지 제어 시스템을 이용하여 학습률을 동적으로 조정하는 퍼지 제어 기법을 이용한 FCM 기반 RBF 네트워크를 제안한다.

제안된 방법의 학습 성능을 평가하기 위하여 컨테이너 영상에서 추출한 숫자, 영문 식별자를 학습 데이터로 적용한 결과, 기존의 ART2 기반 RBF 네트워크보다 학습 시간이 적게 소요되고, 학습의 수렴성이 개선된 것을 확인하였다.

▶ Keyword : FCM 기반 RBF 네트워크, Max\_Min 신경망, 학습률, 퍼지 제어 시스템

---

• 제1저자 : 김태형

\* 신라대학교 컴퓨터정보공학부 \*\*영동대학교 컴퓨터공학과

## I. 서론

ARBF(Radial Basis Function) 네트워크의 중간층은 클러스터링 하는 층이다. 즉, 이 층의 목적은 주어진 자료 집합을 유사한 클러스터들(homogeneous clusters)로 분류하는 것이다. 여기서 유사하다는 것은 입력 데이터들에 대한 특징 벡터 공간사이에서 한 클러스터 내의 벡터들 간에 거리를 측정하여 정해진 반경 내에 존재하면 같은 클러스터로 분류하고 정해진 반경 내에 존재하지 않으면 다른 클러스터로 분류한다[1]. 그러나 정해진 반경 내에서 클러스터링 하는 것은 잘못된 클러스터를 선택하는 단점을 가지게 된다. 그러므로 중간층을 결정하는 것은 RBF 네트워크의 전반적인 효율성에 큰 영향을 준다. RBF 네트워크는 학습이 완료된 상태, 즉 연결 강도가 특정 값으로 고정된 상태에서 새로운 패턴을 학습 시키게 되면 이미 설정된 모든 연결 강도에 영향을 주므로 RBF 네트워크를 다시 학습 시키는데 상당한 학습 시간이 소요되는 단점과 RBF 네트워크에 학습되지 않은 전혀 새로운 형태의 패턴들이 들어오는 경우 이전에 학습된 유사한 패턴으로 분류하여 인식률을 저하시킨다[2]. 따라서 본 논문에서는 FCM 알고리즘을 적용하여 RBF 네트워크의 학습 구조와 성능을 개선한다.

Max-Min 신경망은 퍼지의 기본 연산자를 사용하고 0과 1사이의 값을 입력 패턴으로 적용하여 학습하는 알고리즘이다. 이 알고리즘의 장점은 기존의 단층 알고리즘이 이진 데이터를 학습하는데 중점을 둔 반면에 퍼지 값인 0과 1사이의 값을 학습하는데 효과적으로 학습할 수 있도록 퍼지 논리 연산자를 이용한 것이 큰 장점이다. 하지만 이 알고리즘은 실제계 적용에 있어 패턴의 수가 증가하는 경우에는 학습률과 모멘텀 설정에 따라 학습의 수렴성이 낮아지는 단점이 있다[3,4]. 따라서 FCM 기반 RBF 네트워크에 Max\_Min 신경망을 중간층과 출력층의 학습구조로 적용할 경우에는 학습 파라미터의 설정에 따라서 학습 및 인식 성능이 좌우되는 문제점이 있다. 따라서 본 논문에서는 중간층과 출력층 사이의 학습 구조인 Max\_Min 신경망의 성능을 개선하기 위해 퍼지 제어 기법을 적용하여 학습률을 동적으로 조정하는 방법을 제안하여 FCM 기반 RBF 네트워크에 적용한다.

## II. FCM 기반 퍼지 RBF 네트워크

FCM 기반 RBF 네트워크는 중간층을 생성하기 위하여

퍼지 C-Means 알고리즘을 적용한다. Bezdek이 제안한 퍼지 C-Means 알고리즘은 어떤 패턴이 오직 한 클러스터에만 속한다고 보는 Hard C-Means 알고리즘에 퍼지 이론의 특성을 포함시켜, 복수개의 클러스터에 서로 다른 정도로 속한다고 정의하는 클러스터링 알고리즘이다[5]. 따라서 패턴 인식 분야에 이 알고리즘을 적용할 경우에는 한 패턴의 특징이 복수개의 클러스터에 분산되므로 학습 시간이 많이 소요되는 단점이 있다. 일반적으로 패턴 인식에 있어서 입력 패턴은 이진 패턴을 사용한다. 따라서 0과 1로 구성된 학습 패턴을 퍼지 C-Means 알고리즘을 적용할 경우에는 정확히 입력 패턴들을 분류하기 어렵고 다른 클러스터링 알고리즘에 비해 학습 시간이 많이 소요 된다[6,7] 따라서 본 논문에서는 FCM 기반 RBF 네트워크의 학습 성능을 개선하기 위하여 중간층과 출력층 사이에 학습 구조로 적용되는 Max\_Min 신경망의 연결 강도를 조정하는 과정에서 학습률을 동적으로 조정하여 학습 시간을 단축하고 학습 성능을 개선하기 위하여 퍼지 제어 기법을 적용한다.

## III. 퍼지 제어 기법을 이용한 FCM 기반 RBF 네트워크

본 논문에서 제안한 퍼지 제어 기법을 이용한 FCM 기반 RBF 네트워크는 중간층에 퍼지 C-Means 알고리즘을 적용하는 학습 구조이고 중간층과 출력층 사이의 학습으로는 Max\_Min 신경망을 적용하고 학습 성능을 개선하기 위하여 퍼지 제어 기법을 적용하여 학습률을 동적으로 조정하는 구조이다.

제안된 RBF 네트워크의 학습 방법은 다음과 같이 요약할 수 있다.

1. 입력층과 중간층의 연결 구조는 퍼지 C-Means 알고리즘과 같고, 퍼지 C-Means 알고리즘의 출력층이 제안된 학습 구조의 중간층이다.
2. 중간층과 출력층 사이의 학습 구조로는 Max\_Min 신경망을 적용하여 지도 학습으로 동작한다.
5. 학습 성능을 개선하기 위하여 Max\_Min 신경망에 퍼지 제어 기법을 적용하여 학습률을 동적으로 조정한다. 학습률을 동적으로 조정하는 방법은 목표 벡터와 출력 벡터의 차이에 대한 절대값이 0.1 보다 적으면 정확성으로 분류하고 크면 부정확성으로 분류하여 정확성과 부정확성의 개수를 퍼지 제어 시스템에 적용하여 학습률을 동적으로 조정한다.

제안된 RBF 네트워크의 학습 단계는 다음과 같다.

- 단계 1 : 학습에 사용되는 인수를 초기화 한다.
- 단계 2 : 학습패턴과 목표패턴을 설정한다.
- 단계 3 : 클러스터의 중심값을 식(1)과 같이 계산한다.

$$v_{ij} = \sum_{k=1}^n (\mu_{ik})^m x_{kj} / \sum_{k=1}^n (\mu_{ik})^m \dots\dots\dots (1)$$

식(1)에서  $v_{ij}$  와  $x_{kj}$  는 각각 클러스터의 중심과 입력 벡터이고,  $m$  는 지수 가중치이다.

단계 4 : 중간층의 소속도를 식(2)와 같이 계산한다.

$$\mu_{ik} = 1 / \sum_{y=1}^c \left( \frac{|x_{kj} - v_{ij}|}{|x_{kj} - v_{yj}|} \right)^{2/m-1} \dots\dots\dots (2)$$

단계 5 : 오차 한계치를 식(3)과 같이 검사한다.

$$|U^{(p)} - U^{(p-1)}| \leq \epsilon \dots\dots\dots (3)$$

현재 소속도( $U^{(p)}$ )와 이전 소속도( $U^{(p-1)}$ )의 차이가 오차 한계치( $\epsilon$ ) 보다 작거나 같으면 학습을 종료하고, 그렇지 않으면 단계 2로 가서 학습을 반복한다.

단계 6 : 중간층의 승자 노드( $O_i^*$ )를 식(4)와 같이 계산한다.

$$\begin{aligned} O_i &= \sqrt{v_{ij}^2 - x_{kj}^2} \\ O_i &= \min(O_i) \\ \text{if } O_i &= O_{i^*} \text{ then } O_i = 1 \\ &\text{else } O_i = 0 \end{aligned} \dots\dots\dots (4)$$

단계 7 : 출력층의 출력 벡터( $O_i$ )를 식(5)와 같이 계산한다.

$$\begin{aligned} Net_l &= \vee \{ \{O_1 \wedge w_{l1}\}, \dots, \{O_i \wedge w_{li}\} \} \\ O_l &= Net_l \vee \theta_l \end{aligned} \dots\dots\dots (5)$$

단계 8 : 오류값을 식(6)과 같이 계산한다.

$$error = \frac{1}{2} (O_l - T_l)^2 \dots\dots\dots (6)$$

단계 9 : 단계 8에서 구해진 목표값( $O_l$ )과 출력값( $T_l$ )의 차이를 이용하여 퍼지 제어 기법을 적용하여 학습률

( $\alpha$ )을 동적으로 조정한다.

$$\alpha = \text{Fuzzy}(\text{정확성의 수, 부정확성의 수}) \dots\dots\dots (7)$$

단계 10 : 동적으로 조정된 학습률을 이용하여 연결 가중치( $w_{li}$ )와 바이어스항( $\theta_l$ )을 식(8)과 같이 계산한다.

$$\begin{aligned} w_{li}(n+1) &= w_{li}(n) + \alpha \Delta w_{li}(n+1) + \beta \Delta w_{li}(n) \\ \theta_l(n+1) &= \theta_l(n) + \alpha \Delta \theta_l(n+1) + \beta \Delta \theta_l(n) \\ \text{if } O_l &= w_{li} \text{ then } \Delta w_{li}(n+1) = \Delta w_{li}(n) + 1 \times error \\ &\text{else } \Delta w_{li}(n+1) = \Delta w_{li}(n) + 0 \times error \\ \text{if } O_l &= \theta_l \text{ then } \Delta \theta_l(n+1) = \Delta \theta_l(n) + 1 \times error \\ &\text{else } \Delta \theta_l(n+1) = \Delta \theta_l(n) + 0 \times error \end{aligned} \dots\dots\dots (8)$$

식(8)에서  $\alpha$  는 학습률이고,  $\beta$  는 모멘텀 계수이다.

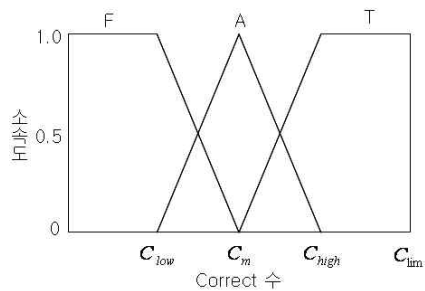
본 논문에서는 Max-Min 신경망의 학습 성능을 개선하기 위해 퍼지 제어 기법을 적용하여 학습률을 동적으로 조정한다. 각 패턴에 대한 실제 출력값과 목표값의 차이가 0.1보다 적거나 같은 경우 정확성으로 분류하고, 큰 경우에는 부정확성으로 분류하여 퍼지 제어 기법의 입력으로 사용한다. 정확성에 대한 소속 함수는 그림 1과 같고 부정확성에 대한 소속 함수는 그림 2와 같다. 그림 1에서 정확성의 소속 함수 구간 [ $C_{low}$ ,  $C_{lim}$ ]는 식(9)에서 식(12)와 같이 계산한다.

$$C_{low} = \log_2(\text{입력 노드 수} + \text{입력 패턴 개수}) \dots\dots (9)$$

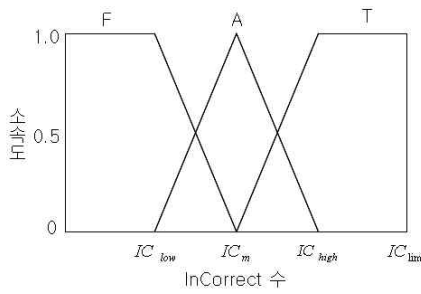
$$C_m = (C_{high} + C_{low}) / 2 \dots\dots\dots (10)$$

$$C_{high} = C_{lim} - C_{low} \dots\dots\dots (11)$$

$$C_{lim} = \text{출력 노드 수} \times \text{입력 패턴 개수} \dots\dots (12)$$

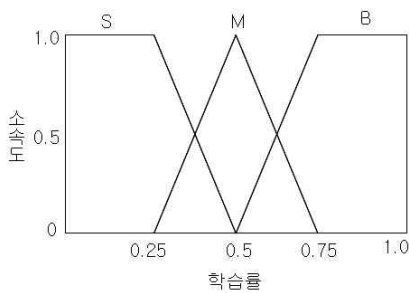


<그림 1> 정확성에 대한 소속 함수



<그림 2> 부정확성에 대한 소속 함수

그림 3은 학습에 적용될 학습률을 계산하는 소속 함수이다.



<그림 3> 학습률 출력 소속 함수

제안된 학습률을 조정하기 위한 퍼지 제어 규칙을 if-then 형식의 퍼지 관계로 나타내면 다음과 같다.

- $R_1$ : If correct is F, Incorrect is F Then  $\alpha$  is B
- $R_2$ : If correct is F, Incorrect is A Then  $\alpha$  is B
- $R_3$ : If correct is F, Incorrect is T Then  $\alpha$  is B
- $R_4$ : If correct is A, Incorrect is F Then  $\alpha$  is M
- $R_5$ : If correct is A, Incorrect is A Then  $\alpha$  is M
- $R_6$ : If correct is A, Incorrect is T Then  $\alpha$  is M
- $R_7$ : If correct is T, Incorrect is F Then  $\alpha$  is S
- $R_8$ : If correct is T, Incorrect is A Then  $\alpha$  is S
- $R_9$ : If correct is T, Incorrect is T Then  $\alpha$  is S

정확성과 부정확성의 개수가 퍼지 제어 기법의 입력값으로 정해지면 각 소속 함수에 대한 정확성과 부정확성의 소속도를 구한다. 각 소속 함수에 대하여 소속도가 구해지면 퍼지 제어 규칙을 적용하고 Mandani의 Min-Max 방법으로 추론한다. 퍼지 추론 후에는 비퍼지화하여 학습에 사용될 학습률

을 구한다. 비퍼지화는 무게 중심법을 적용한다.

### IV. 실험 및 결과 분석

본 논문에서 제안된 방법의 학습 및 인식 성능을 평가하기 위하여 Intel Pentium-IV 3.0GHz CPU와 1GB RAM이 장착된 IBM 호환 PC상에서 Visual Studio 6.0으로 구현하였다. 제안된 방법과 ART2 기반 RBF 네트워크간의 학습 및 인식 성능을 비교하기 위해 컨테이너에서 추출한 숫자 및 영문 식별자를 대상으로 실험하였다. 표 1은 제안된 방법에 사용된 학습 파라미터이다.

<표 1> 제안된 방법에 사용된 학습 파라미터

	$c$	$m$	$\epsilon$	$\beta$
숫자	10	10	0.01	0.7
영문	24	1000	0.01	0.7

표 1에서  $c$ 는 클러스터,  $m$ 는 지수 가중치,  $\epsilon$ 는 오류 한계치,  $\beta$ 는 모멘텀 계수를 나타내는 파라미터 이다.

표 2은 ART2기반 RBF 네트워크에 사용된 파라미터이다.

<표 2> ART2기반 RBF 네트워크에 사용된 학습 파라미터

	$\rho$	$\gamma$	$k$	$\beta$	$\epsilon$
숫자	0.14	0.3	0.05	0.7	0.01
영문	0.1	0.3	0.05	0.7	0.01

표 2에서  $\rho$ 는 ART2 알고리즘의 경계변수이고,  $\gamma$ ,  $k$ ,  $\beta$ 는 delta-bar-delta 상수이고,  $\epsilon$ 는 오류한계치 이다. ART2 기반 RBF 네트워크는 ART2의 단점인 경계변수 설정에 따라 중간층의 노드의 수가 증가하거나 감소하였다. 따라서 표 2에서와 같이 경계 변수  $\rho$ 는 다양한 실험을 통해서 숫자 패턴에서는 0.14, 영문 패턴에서는 0.1로 설정하는 것이 최적으로 나타났다.

표 3과 4는 컨테이너 영상에서 추출한 숫자와 영문 패턴에 대해 제안된 방법과 ART2 기반 RBF 네트워크 간의 학습 및 인식 성능을 각각 비교한 결과이다.

숫자 패턴은 0에서 9까지 각각 30개의 패턴을 학습에 사용하였고, 영문 패턴은 Q와 V를 제외한 24개의 영문 패턴들을 추출하고 각 추출한 영문 패턴 20개씩을 학습에 사용되었다.

<표 3> 제안된 방법과 ART2기반 RBF 네트워크의 숫자 식별자에 대한 학습 및 인식결과

	제안된 방법	ART2 기반 RBF 네트워크
클러스터 수	10	48
Epoch 수	2	85864
인식률	300/300	300/300
TSS	0	0.01

<표 4> 제안된 방법과 ART2기반 RBF 네트워크의 영문 식별자에 대한 학습 및 인식결과

	제안된 방법	ART2 기반 RBF 네트워크
클러스터 수	24	173
Epoch 수	2	100000
인식률	480/480	480/480
TSS	0	1.005729

표 3과 표 4에서 확인할 수 있듯이 제안된 방법이 ART2 기반 RBF 네트워크보다 학습 속도나 학습의 수렴성에 있어서 개선된 것을 확인할 수 있다. 또한 제안된 FCM 기반 RBF 네트워크에서는 퍼지 C-Means 알고리즘을 중간층으로 적용하므로 중간층의 노드의 소속도에 따라서 클러스터를 생성하므로 ART2 기반 RBF 네트워크보다 중간층의 노드가 적게 생성되었으나 숫자 패턴과 영문 패턴들이 모두 인식되어 패턴 인식이 효율적인 것을 알 수 있다. 제안된 FCM 기반 RBF 네트워크로 숫자와 영문 식별자를 각각 학습한 결과, 학습의 반복 횟수가 2회로 나타났다.

그 이유는 입력층과 중간층의 학습 구조로 FCM 알고리즘을 적용하여 입력 벡터들의 특징이 중복 되는 부분을 최소화하여 중간층에서 클러스터를 생성하고, Max\_Min 신경망이 적용되는 중간층과 출력층 사이의 학습 과정에서 조기 포화현상(premature situation)이 발생하여 오류가 거의 일정하게 유지되면서 학습이 진전되지 않거나 진동하는 부분을 방지하기 위해 목표 벡터와 출력 벡터의 차이가 0.1 보다 적으면 정확성으로 정의하고, 크면 부정확성으로 정의하여 정확성과 부정확성의 개수를 퍼지 제어 시스템에 적용하여 학습률을 동적으로 조정하여 조기포화현상을 방지하였기 때문이다.

그림 4는 숫자 패턴에 대한 제안된 방법과 ART2 기반 RBF 네트워크의 학습 반복 횟수에 따른 TSS 변화 과정을

나타내었다. 그림 4에서 알 수 있듯이 제안된 방법은 조기포화 현상이 일어나지 않는 것을 알 수 있고 ART2 기반 RBF 네트워크에서는 학습의 반복 횟수가 약 10000회 이상에서는 조기 포화 현상이 발생하는 것을 알 수 있다.



<그림 4> 반복 횟수에 따른 TSS 변화 과정

### V. 결론

RBF 네트워크의 중간층은 학습 패턴들을 클러스터링하는 층이다. 이 중간층의 목적은 주어진 학습 패턴들을 유사한 클러스터들로 분류하는 것이다. 여기서 유사하다는 것은 학습 패턴들에 대한 특징 벡터 공간 사이에서 한 클러스터내의 벡터들 간에 거리를 측정하여 정해진 반경 내에 존재하면 같은 클러스터에 속하고 정해진 반경 내에 존재하지 않으면 다른 클러스터에 속한다. 정해진 반경 내에서 클러스터링 하는 것은 잘못된 클러스터를 선택하여 출력층의 입력값으로 구성하므로 RBF 네트워크의 학습 성능을 저하시킨다. 또한 학습 과정에서 중간층의 노드는 그 노드의 목표 벡터를 알 수 없기 때문에 출력층 노드의 오차를 역으로 중간층 노드들로 책임 분담(credit assignment)하는 방법을 사용하여 정체현상이 일어나므로 학습 시간이 많이 소요된다. 따라서 본 논문에서는 중간층에 퍼지 C-Means 알고리즘을 적용하여 소속도가 가장 큰 노드를 승자 노드로 선택하여 출력층으로 전달하는 학습 구조를 제안하였고 중간층과 출력층 사이의 학습 구조는 Max\_Min 알고리즘을 적용하였다. Max\_Min 신경망을 제안된 RBF 네트워크의 중간층과 출력층 사이의 학습 구조로 사용할 경우에는 학습률과 같은 파라미터에 의해 학습의 수렴성이 낮아지는 경우가 발생하고 학습 시간이 많이 소요되는 경우가 발생한다. 따라서 이러한 단점을 개선하기 위해 본 논문에서는 퍼지 제어 기법을 적용하여 학습률을 동적으로 조정하여 조기포화현상으로 학습의 수렴성이 낮아지거나 학습 시간이 많이 소요되는 부분을 개선하였다. 제안된 학습률 동적 조정 방법은, 출력 벡터와 목표 벡터의 차이의 절대값이 0.1보다 적거나 같으면 정확성으로 분류하고, 그렇지 않은 경우에

는 부정확성으로 분류하였다. 그리고 정확성과 부정확성의 수를 퍼지 제어 기법을 적용하여 학습률을 동적으로 조정하였고 동적으로 조정된 학습률을 이용해 연결가중치와 바이어스항을 조정하였다.

제안된 학습 방법의 학습 및 인식 성능을 평가하기 위해 컨테이너에서 추출된 숫자와 영문 식별자를 학습 및 인식 성능에 적용한 결과, Delta-bar-Delta 알고리즘을 이용하여 학습률을 동적으로 조정하는 ART2 기반 RBF 네트워크 보다 중간층의 클러스터가 적게 생성되었고 학습의 수렴성과 학습 속도가 개선되었다. 또한 인식 성능에 있어서도 우수한 성능을 보임을 확인하였다.

향후 연구 과제는 제안된 RBF 네트워크를 다양한 패턴 인식 분야에 적용하고 퍼지 신경망 알고리즘뿐만 아니라 다양한 인공지능 알고리즘과 학습 및 인식 성능을 비교 분석할 것이다.

## 참고문헌

- [1] M. L. Kothari, S. Machani and R. Segal, "Orthogonal Least Square Learning Algorithm Based Radial Basis Function Network Adaptive Power System Stabilizer." Proceeding of IEEE SMC, Vol.1 pp.542-547, 1997
- [2] 김광백, "퍼지 RBF 네트워크의 학습 성능 개선," 멀티미디어학회논문지, 9권 3호, pp.369-376, 2006.
- [3] 김광백, 박현정, "개선된 Max-Min 신경망을 이용한 콘크리트 균열 인식," 한국컴퓨터정보학회논문지, 12권 2호, pp.77-82, 2007.
- [4] K. B. Kim, J. H. Cho, A. S. Oh, " Recognition of English Business Cards Using Enhanced Hybrid Network," Lecture Notes in Computer Science, LNCS 3497, Springer, pp.209-215, 2005.
- [5] R. Babuska, Fuzzy Modeling for Control, Kluwer Academic Publishers, 1998.
- [6] K. B. Kim, Y. H. Joo, J. H. Cho, "An Enhanced Fuzzy Neural Network," Lecture Notes in Computer Science, LNCS 3320, pp.176-179, 2004.
- [7] K. B. Kim and C. K. Kim, "Performance Improvement of RBF Network using ART2 Algorithm and Fuzzy Logic System" Lecture Notes in Artificial Intelligence, LNAI 3339, pp.853-860, 2004.