

論文2002-39CI-1-5

TSK퍼지시스템을 결론부가 singleton인 퍼지시스템으로 표현하는 방법과 그 응용

(Transformation of TSK fuzzy systems into fuzzy systems with singleton consequents and its applications)

蔡良範*, 李遠昌**, 姜根澤**

(Yangbum Chae, Geuntaek Kang, and Wonchang Lee)

요 약

본 논문에서는 어느 한 TSK(Takagi-Sugeno-Kang) 퍼지시스템이 주어 졌을 때 그 퍼지시스템과 동일한 입출력 관계를 갖는 singleton 퍼지시스템을 구하는 방법을 제안하고 응용 예를 보인다. 퍼지규칙의 결론부가 선형식인 퍼지시스템(TSK퍼지시스템)은 입출력 데이터로 모델 인식이 체계적으로 쉽게 이루어 질 수 있으며, 안정성을 보장하는 퍼지제어기 설계도 관한 연구도 많이 되어 있다. 한편 퍼지규칙 결론부가 실수인 퍼지시스템(singleton 퍼지시스템)은 규칙이 언어적 형태이므로 이해하기가 쉽고, 규칙의 조정이 용이한 장점이 있다. 이러한 두 퍼지 시스템의 장점을 살릴 수 있는 방안으로, TSK 퍼지시스템을 singleton 퍼지시스템으로 변환시키는 방법을 제안하며, 제안한 방법을 퍼지모델링과 퍼지제어기 설계에 응용하여 그 실용성을 보인다.

Abstract

TSK(Takagi-Sugeno-Kang) fuzzy models with linear equations consequents, which represent complex nonlinear systems very well with a few rules, can be easily identified systematically by using input-output data. Many algorithms designing TSK fuzzy controllers based on TSK fuzzy models, which guarantees the stability of the closed system, have been suggested. On the contrary, singleton fuzzy models with singleton consequents can be easily understood and adjusted. In this paper, in order to utilize the merits of TSK fuzzy systems and singleton fuzzy systems, an algorithm transforming a TSK fuzzy model into a singleton fuzzy model having the same input-output relation is suggested. The suggested algorithm is applied to a fuzzy modelling example and a fuzzy controller design example.

I. 서 론

* 正會員, 韓國海洋大學校 운항시스템공학부
(Division of ship Operation Systems Engineering
Korea Maritime University)

** 正會員, 釜慶大學校 電子컴퓨터情報通信工學部
(Division of Electronic, Computer and Telecommunication Engineering Pukyong National University)
接受日字:2001年7月11日, 수정완료일:2001年12月17日

퍼지 이론이 처음 발표된 후 많은 분야에서 적용되어 연구되어 왔으며, 특히 Zadeh가 퍼지이론을 토대로 퍼지 모델링을 발표한 후^[1] if - then 형태의 퍼지규칙으로 구성되는 퍼지시스템이 많이 사용되고 있다. 퍼지시스템은 퍼지규칙 결론부의 형태 (퍼지집합, 실수값, 입출력 변수의 선형식)에 따라 나뉜다.

퍼지규칙의 결론부가 선형식인 퍼지시스템^[2] (이하

TSK퍼지시스템라 한다.)은 적은 수의 퍼지규칙으로 복잡한 시스템을 잘 표현 할 수 있으며^[3,4], 입출력 데이터로 모델 인식이 체계적으로 쉽게 이루어 질 수 있다^[5~8]. 또한 결론부가 선형식인 점을 이용하여 안정성을 보장하는 퍼지제어기 설계에 관한 연구도 많이 되어 있다^[9~12].

한편 퍼지규칙 결론부가 실수인 형태(이하 singleton 퍼지시스템라 한다.)는 TSK퍼지 시스템을 간략화한 형태로 규칙이 언어적 형태이므로 이해하기가 쉽고, 규칙의 조정이 용이한 장점이 있다.

따라서 TSK 퍼지시스템과 singleton 퍼지시스템의 두 퍼지 시스템의 장점을 살릴 수 있는 방안이 필요하며, 그 방안으로 TSK퍼지시스템을 singleton 퍼지시스템으로 변환시키는 방법이 발표되고있으나^[13,14] 두 시스템이 동일한 입출력 관계를 갖는 변환 방법은 아직 발표되지 않고 있다. 이에 본 논문에서는 어느 한 TSK 퍼지시스템이 주어 졌을 때 그 퍼지시스템을 동일한 입출력 관계를 갖는 singleton 퍼지시스템으로 변환시키는 방법을 제안하며, 제안한 방법을 퍼지모델링과 퍼지제어기 설계에 응용할 수 있음을 보인다.

II. TSK 퍼지 모델을 singleton 퍼지모델로 변환

이 장에서는 TSK퍼지모델과 동일한 입출력 관계를 갖는 singleton 퍼지모델을 구하는 방법을 설명한다. TSK 퍼지모델의 k 번째 규칙이 다음과 같다 하자

$$R^k : \text{If } x \text{ is } A^k \text{ then } y^k = a_0^k + (a^k)^T x \quad (1)$$

여기서 $x = (x_1 \ x_2 \ \dots \ x_n)^T$ 는 입력 변수 벡터, $A^k = (A_1^k \ A_2^k \ \dots \ A_n^k)^T$ 는 전제부 퍼지집합들의 벡터, y^k 는 출력 변수, $a^k = (a_1^k \ a_2^k \ \dots \ a_n^k)^T$ 는 결론부 파라미터 벡터이며, a_0^k 는 결론부의 상수항이다. 퍼지모

델이 r 개의 퍼지규칙들로 구성될 경우, 그 퍼지모델로부터 출력 y 는 다음 식과 같이 추론된다.

$$y = \frac{\sum_{k=1}^r A^k(x) y^k}{\sum_{k=1}^r A^k(x)} \quad (2)$$

여기서 $A^k(x)$ 는 k 번째 규칙의 적합도를 나타내며 x_i 에서 퍼지집합 A_i^k 의 멤버십 값을 $A_i^k(x)$ 라하면 다음과 같다.

$$A^k(x) = \prod_{i=1}^n A_i^k(x_i) \quad (3)$$

식 (1)과 같은 퍼지규칙들로 구성되는 TSK 퍼지모델은 시스템의 입출력 데이터를 이용하여 만들어진다^[6]. TSK 퍼지모델 인식 때 맨 먼저 정해야 할 것은 전제부 퍼지집합의 형태이다. 전제부의 퍼지집합의 멤버십 함수로는 여러 형태가 사용되고 있으나 본 논문에서는 멤버십 함수가 간단한 선형식인 그림 1과 같은 사다리꼴을 사용하도록 한다. 또한 그림 1과 같이 한 퍼지집합의 멤버십 값이 0으로 시작되는 점이 옆의 퍼지집합의 멤버십 값이 1로 시작되는 점과 일치하도록 하여 (가령 그림 1에서 μ_1 은 퍼지집합 F_1 에서는 멤버십 값이 1로 시작되는 점이면서 퍼지집합 F_2 에서는 멤버십 값이 0으로 시작되는 점이다.) 모델을 간단한 형태로 하였다. 한편 위와 같이 퍼지집합을 만들 경우 식 (2)에서 $\sum_{k=1}^r A^k(x) = 1$ 가 만족함을 쉽게 알 수 있다. 따라서 본 논문에서는 식 (2)의 분모는 항상 1로 한다. 이와 같은 제한은 시스템의 입출력 데이터를 이용한 TSK 퍼지 모델링 단계에서 나온 것이므로 일반성에는 문제가 없다고 본다. 그림 1에서 μ_S 와 μ_L 는 퍼지 모델이 적용 가능한 변수의 최소 값 및 최대 값이다.

위의 TSK 퍼지모델을 singleton 퍼지모델로 변환은, 식(1)의 퍼지규칙의 결론부에서 변수 x_1, x_2, \dots, x_n

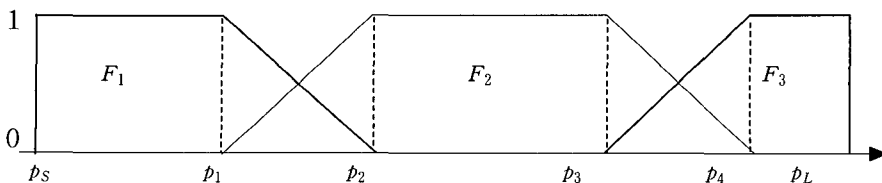


그림 1. TSK 퍼지모델에서 사용되는 퍼지집합
Fig. 1. Fuzzy sets in TSK fuzzy models.

를 하나씩 제거하여 상수항만 남게 함으로써 이루어지며, 변수를 하나씩 제거 할 때마다 퍼지 규칙의 수는 증가한다.

변수를 제거하는 알고리즘은 다음의 기본 아이디어를 사용한다. 즉, 식 (4)와 같은 TSK 퍼지규칙을 singleton 퍼지모델로 표현하면 그림 2와 같아짐을 이용한다.

$$\text{If } x \text{ is } F \text{ then } y = a_0 + a_1x \quad (4)$$

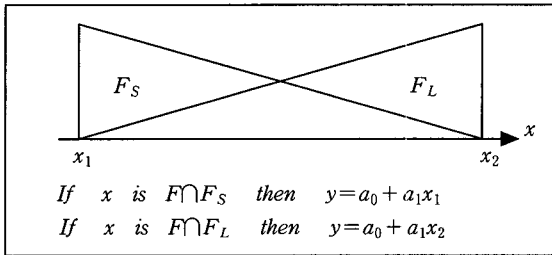


그림 2. 식(4)의 TSK퍼지규칙과 동일한 singleton 퍼지모델

Fig. 2. Singleton fuzzy model of the TSK fuzzy rule (4).

그림 2 퍼지모델의 출력은 $(F \cap F_S)(x) \cdot (a_0 + a_1x_1) + (F \cap F_L)(x) \cdot (a_0 + a_1x_2)$ 이며 식(4)의 퍼지규칙의 출력은 $F(x) \cdot (a_0 + a_1x)$ 이다. 식(5)에서 그 두 결과가 동일함을 보여준다. 식(3)의 규칙 적합도 계산에서 곱셈을 사용하므로 그림 2 및 식 (5)에서 퍼지집합의 intersection은 곱셈을 이용한다. (즉, $(F \cap F_S)(x) = F(x) \cdot F_S(x)$)

$$\begin{aligned} & (F \cap F_S)(x) \cdot (a_0 + a_1x_1) + (F \cap F_L)(x) \cdot (a_0 + a_1x_2) \\ &= F(x) \cdot (F_S(x) \cdot (a_0 + a_1x_1) + F_L(x) \cdot (a_0 + a_1x_2)) \quad (5) \\ &= F(x) \cdot \left(\frac{x_2 - x}{x_2 - x_1} \cdot (a_0 + a_1x_1) + \frac{x - x_1}{x_2 - x_1} \cdot (a_0 + a_1x_2) \right) \\ &= F(x) \cdot (a_0 + a_1x) \end{aligned}$$

TSK 퍼지모델의 결론부에서 변수를 하나씩 제거하는 알고리즘은, 전제부에 그 변수가 있는지, 만약 있

면 그 퍼지집합의 형태가 그림 1의 F_1, F_2, F_3 중 어느 형태인가에 따라 틀린다.

지금부터 식 (1)의 퍼지규칙 결론부에서 변수 x_i 를 제거하는 알고리즘을 4개의 case로 나누어서 설명한다. 아래의 알고리즘에서 $x^- = (x_1 \ x_2 \ \dots \ x_{i-1} \ x_{i+1} \ \dots \ x_n)^T$, $A^{k-} = (A_1^k \ A_2^k \ \dots \ A_{i-1}^k \ A_{i+1}^k \ \dots \ A_n^k)^T$, $a^{k-} = (a_1^k \ a_2^k \ \dots \ a_{i-1}^k \ a_{i+1}^k \ \dots \ a_n^k)^T$ 로 정의하며, 알고리즘에 나오는 퍼지집합 $B_1 \sim B_8$ 과 그 멤버십 함수는 그림3과 같다. 퍼지집합 $B_1 \sim B_8$ 은 그림 1의 5개 구간 $[p_5 \ p_1], [p_1 \ p_2], [p_2 \ p_3], [p_3 \ p_4], [p_4 \ p_L]$ 에 대해 그림 2와 같은 작업을 하여 얻어진 퍼지집합들이다. 그림 3에서 퍼지집합 B_3 과 B_6 은 원래 최고점의 멤버십 값이 0.25이지만 normalize시켜 1로 만들었다. 그 대신 B_3 과 B_6 에 해당하는 규칙의 결론부 실수값은 $\frac{1}{4}$ 배로 줄어든다. (식(6), (9), (15), (16) 참조)

$$\begin{aligned} B_1(x_i) &= \begin{matrix} (p_1 - x_i)/(p_1 - p_5) & : & p_5 \leq x_i \leq p_1 \\ 0 & : & \text{otherwise} \end{matrix} \\ B_2(x_i) &= \begin{matrix} (x_i - p_5)/(p_1 - p_5) & : & p_5 < x_i \leq p_1 \\ (p_2 - x_i)^2/(p_2 - p_1)^2 & : & p_1 \leq x_i \leq p_2 \\ 0 & : & \text{otherwise} \end{matrix} \\ B_3(x_i) &= \begin{matrix} 4(p_2 - x_i)(x_i - p_1)/(p_2 - p_1)^2 & : & p_1 < x_i < p_2 \\ 0 & : & \text{otherwise} \end{matrix} \\ B_4(x_i) &= \begin{matrix} (x_i - p_1)^2/(p_2 - p_1)^2 & : & p_1 \leq x_i \leq p_2 \\ (p_3 - x_i)/(p_3 - p_2) & : & p_2 < x_i \leq p_3 \\ 0 & : & \text{otherwise} \end{matrix} \\ B_5(x_i) &= \begin{matrix} (x_i - p_2)/(p_3 - p_2) & : & p_2 < x_i \leq p_3 \\ (p_4 - x_i)^2/(p_4 - p_3)^2 & : & p_3 \leq x_i \leq p_4 \\ 0 & : & \text{otherwise} \end{matrix} \\ B_6(x_i) &= \begin{matrix} 4(p_4 - x_i)(x_i - p_3)/(p_4 - p_3)^2 & : & p_3 < x_i < p_4 \\ 0 & : & \text{otherwise} \end{matrix} \\ B_7(x_i) &= \begin{matrix} (x_i - p_3)^2/(p_4 - p_3)^2 & : & p_3 \leq x_i \leq p_4 \\ (p_L - x_i)/(p_L - p_4) & : & p_4 < x_i \leq p_L \\ 0 & : & \text{otherwise} \end{matrix} \\ B_8(x_i) &= \begin{matrix} (x_i - p_4)/(p_L - p_4) & : & p_4 < x_i \leq p_L \\ 0 & : & \text{otherwise} \end{matrix} \end{aligned}$$

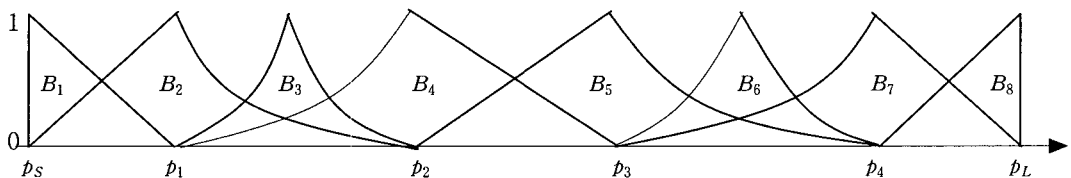


그림 3. 퍼지집합
Fig. 3. Fuzzy sets.

case 1 : 식 (1)에서 x_i 의 퍼지집합 A_i^k 가 그림 1의 F_2 일 때

식 (1)의 R^k 퍼지규칙은 다음 4개 퍼지규칙으로 대체된다.

R^{k1} : If x^- is A^{k-} , x_i is B_3
 then $y^{k1} = \frac{a_0^k + p_1 a_i^k}{4} + \frac{(a^{k-})^T}{4} x^-$ (6)

R^{k2} : If x^- is A^{k-} , x_i is B_4
 then $y^{k2} = a_0^k + p_2 a_i^k + (a^{k-})^T x^-$ (7)

R^{k3} : If x^- is A^{k-} , x_i is B_5
 then $y^{k3} = a_0^k + p_3 a_i^k + (a^{k-})^T x^-$ (8)

R^{k4} : If x^- is A^{k-} , x_i is B_6
 then $y^{k4} = \frac{a_0^k + p_4 a_i^k}{4} + \frac{(a^{k-})^T}{4} x^-$ (9)

식 (1)의 퍼지 규칙 R^k 을 식(6)~(9)의 $R^{k1} \sim R^{k4}$ 4개의 퍼지 규칙들로 대체할 경우 퍼지모델로부터 추론되는 결과는 다음과 같다.

$$y = \sum_{j=1}^{k-1} A^j(x) y^j + \sum_{j=1}^4 A^{k-}(x^-) B_{j+2}(x_i) y^{kj} + \sum_{j=k+1}^r A^j(x) y^j \quad (10)$$

따라서 식(6)~(9)의 $R^{k1} \sim R^{k4}$ 4개의 퍼지 규칙들로 대체한 퍼지모델의 출력인 식(10)이 대체하기 전 원래의 퍼지모델의 출력인 식(2)와 동일함을 보이려면 다음 식이 만족함을 증명하면 된다. (단 식 (2)에서

$$\sum_{k=1}^r A^k(x) = 1$$

$$A^k(x) y^k = A^{k-}(x^-) B_3(x_i) y^{k1} + A^{k-}(x^-) B_4(x_i) y^{k2} + A^{k-}(x^-) B_5(x_i) y^{k3} + A^{k-}(x^-) B_6(x_i) y^{k4} \quad (11)$$

증명 :

먼저 그림 1에서 $[p_1 \ p_2]$ 구간에 대해 증명한다. 이 구간에서는 변환된 규칙들 중에서 식 (6)과 (7)의 2개의 퍼지 규칙이 적용되어 다음과 같이 증명된다.

$$A^{k-}(x^-) B_3(x_i) y^{k1} + A^{k-}(x^-) B_4(x_i) y^{k2} + A^{k-}(x^-) B_5(x_i) y^{k3} + A^{k-}(x^-) B_6(x_i) y^{k4}$$

$$= A^{k-}(x^-) (B_3(x_i) (\frac{a_0^k + p_1 a_i^k}{4} + \frac{(a^{k-})^T}{4} x^-) + B_4(x_i) (a_0^k + p_2 a_i^k + (a^{k-})^T x^-))$$

$$= A^{k-}(x^-) \frac{x_i - p_1}{p_2 - p_1} (\frac{p_2 - x_i}{p_2 - p_1} (a_0^k + p_1 a_i^k + (a^{k-})^T x^-) + \frac{x_i - p_1}{p_2 - p_1} (a_0^k + p_2 a_i^k + (a^{k-})^T x^-))$$

$$= A^{k-}(x^-) A_i^k(x_i) (a_0^k + a_i^k x_i + (a^{k-})^T x^-)$$

$$= A^k(x) (a_0^k + (a^k)^T x)$$

$$= A^k(x) y^k \quad (12)$$

나머지 구간 $[p_2 \ p_3]$ 과 $[p_3 \ p_4]$ 에 대해서도 식(11)이 만족됨을 같은 방법으로 증명할 수 있다. ■

case 2 : 식 (1)에서 x_i 의 퍼지집합 A_i^k 가 그림 1의 F_1 일 때

식 (1)의 R^k 퍼지규칙은 다음 3개 퍼지규칙으로 대체된다.

R^{k1} : If x^- is A^{k-} , x_i is B_1
 then $y^{k1} = a_0^k + p_5 a_i^k + (a^{k-})^T x^-$ (13)

R^{k2} : If x^- is A^{k-} , x_i is B_2
 then $y^{k2} = a_0^k + p_1 a_i^k + (a^{k-})^T x^-$ (14)

R^{k3} : If x^- is A^{k-} , x_i is B_3
 then $y^{k3} = \frac{a_0^k + p_2 a_i^k}{4} + \frac{(a^{k-})^T}{4} x^-$ (15)

case 3 : 식 (1)에서 x_i 의 퍼지집합 A_i^k 가 그림 1의 F_3 일 때

식 (1)의 R^k 퍼지규칙은 다음 3개 퍼지규칙으로 대체된다.

R^{k1} : If x^- is A^{k-} , x_i is B_6
 then $y^{k1} = \frac{a_0^k + p_3 a_i^k}{4} + \frac{(a^{k-})^T}{4} x^-$ (16)

R^{k2} : If x^- is A^{k-} , x_i is B_7
 then $y^{k2} = a_0^k + p_4 a_i^k + (a^{k-})^T x^-$ (17)

R^{k3} : If x^- is A^{k-} , x_i is B_8
 then $y^{k3} = a_0^k + p_L a_i^k + (a^{k-})^T x^-$ (18)

case 4 : 식 (1)에서 전체부에 변수 x_i 가 없을 때

식 (1)의 R^k 퍼지규칙은 다음 2개 퍼지규칙으로 대체된다.

R^{k1} : If x^- is A^{k-} , x_i is B_5
 then $y^{k1} = a_0^k + p_5 a_i^k + (a^{k-})^T x^-$ (19)

$$R^k: \text{If } x^- \text{ is } A^{k-}, x_i \text{ is } B_L \\ \text{then } y^k = a_0^k + p_L a_i^k + (a^{k-})^T x^- \quad (20)$$

위 식의 퍼지집합 B_S 와 B_L 은 다음과 같다.

$$B_S(x_i) = \begin{cases} (p_L - x_i)/(p_L - p_S) & : p_S \leq x_i \leq p_L \\ 0 & : \text{otherwise} \end{cases} \\ B_L(x_i) = \begin{cases} (x_i - p_S)/(p_L - p_S) & : p_S \leq x_i \leq p_L \\ 0 & : \text{otherwise} \end{cases}$$

case 2~case 4 에서도 식 (1)의 퍼지 규칙 R^k 을 새로운 퍼지 규칙들로 대체하여도 퍼지모델로부터 추론되는 결과는 변화가 없으며 그것은 case 1에서 설명한 방법으로 증명된다.

위에서 설명한 알고리즘으로 TSK 퍼지모델에서 결론부 변수를 하나 하나씩 제거하면 결론부에 상수항만 남게되어 최종적으로 singleton 퍼지모델로 변환되어지며, 그 singleton 퍼지모델은 원래의 TSK퍼지모델과 동일한 입출력 관계를 갖는다. 변환된 singleton 퍼지모델에서 전제부가 같은 규칙들이 나올 수 있으며 (예로서 식 (6)과 (15), 식(9)와 (16)은 전제부가 같다.) 그러한 규칙들은 결론부의 상수 값을 합하여 하나의 퍼지 규칙으로 만든다.

최종에 singleton 퍼지모델로 변환되었을 때 퍼지 규칙의 개수는 다음과 같다. TSK 퍼지모델에서 변수 x_i 의 구간이 N_i 개로 나뉘어져 있을 때 (그림 1의 경우는 3개로 나뉘어져 있는 경우) singleton 퍼지모델에서는 변수 x_i 의 구간이 $2+3(N_i-1)$ 개로 나뉘어 진다. TSK퍼지모델의 전제부에 변수 x_i 가 없을 경우는 $N_i=1$ 로 한다. 따라서 변수 개수가 n 개인 경우 TSK 퍼지모델의 규칙 개수는 최대 $\prod_{i=1}^n N_i$ 개이고 변환된 singleton 퍼지모델의 규칙 개수는 최대 $\prod_{i=1}^n (2+3(N_i-1))$ 개이다.

III. 퍼지 모델링에 응용

퍼지 모델링의 수법으로는, 시스템의 입출력을 이용한 방법, 시스템의 전문가의 언어적인 지식을 이용하는 방법으로 나눌 수 있다. TSK퍼지 모델의 경우 거의 전적으로 시스템의 입출력 데이터 또는 시스템의 거동을 기술하는 수식을 바탕으로 모델링한다. 그에 반해 singleton 퍼지 모델의 경우, 대부분 모델의 대상 시스

템 전문가의 언어적 지식으로 바탕으로 모델링 된다.

시스템의 입출력 데이터로 퍼지모델을 인식해야 할 경우 가장 문제가 되는 것은 모델의 구조 인식이다. 퍼지모델에서 구조 인식이란 전제부를 어떻게 구성하는가를 결정하는 문제이다. 즉 전제부에 사용되는 변수들과 그 변수들로 구성되는 공간을 어떻게 퍼지 분할 할 것인가를 결정하는 것이 전제부 인식이다. 이러한 전제부 인식이 이루어지면 결론부 인식은 선형시스템의 파라미터 인식 수법을 그대로 이용하여 쉽게 이뤄진다.

시스템의 입출력 데이터를 이용한 TSK 퍼지모델의 인식은 쉽게 이뤄진다. 왜냐하면 TSK 퍼지모델은 적은 수의 규칙으로 복잡한 비선형 시스템을 정도 높게 표현 할 수 있어 모델의 구조 인식이 체계적으로 비교적 쉽게 이루어질 수 있다. 이에 반해 singleton 퍼지모델의 경우, 시스템이 복잡하고 입력 변수가 많아지면 규칙의 수가 기하 급수적으로 많아져 입출력 데이터를 이용한 모델의 구조인식이 어렵다.

한편 TSK 퍼지모델은 결론부가 선형식이므로 결론부가 실수인 singleton 퍼지모델에 비해 모델의 이해가 어려우며 필요한 부분만을 손으로 수정하기도 어렵다. 즉, 신경회로망 또는 GA 등을 이용한 기존의 TSK 퍼지모델의 인식 기법들은 입출력 데이터를 이용하여 데이터 전체 오차의 분산과 같은 어느 한 목적 함수를 최소로 하도록 한다. 그러나 입력공간의 어느 한 특성의 부분에서 모델을 수정해야할 경우 TSK 퍼지모델의 형태로는 규칙의 수정이 어렵다. 그에 반해 시스템을 singleton 퍼지모델로 표현하면 operator가 필요에 따라 수정이 필요한 부분을 손쉽게 수동으로 수정할 수 있다.

따라서 이 장에서는 복잡한 비선형시스템에 대해 입출력 데이터만 주어 졌을 때, 그 데이터를 이용하여 TSK퍼지모델로 인식하고, 그 모델을 앞장에서 설명한 방법으로 singleton 퍼지모델로 변환시키고 필요한 부분을 수동으로 수정할 수 있음을 예를 통해 보여준다.

반도체 소자 모델링 예 :

강한 비선형성과 복잡한 동특성을 갖는 반도체 소자들 중에는 물리적 해석에 의한 모델을 얻기가 어렵거나 사용하기에는 너무 복잡한 것들이 있다. 그러한 소자들의 모델링에는 black-box적인 방법들이 많이 사용되며, 그러한 수법으로는 look-up 테이블, 스플라인 함수, 구분선형함수 등이 있다. 그러나 look-up 테이블

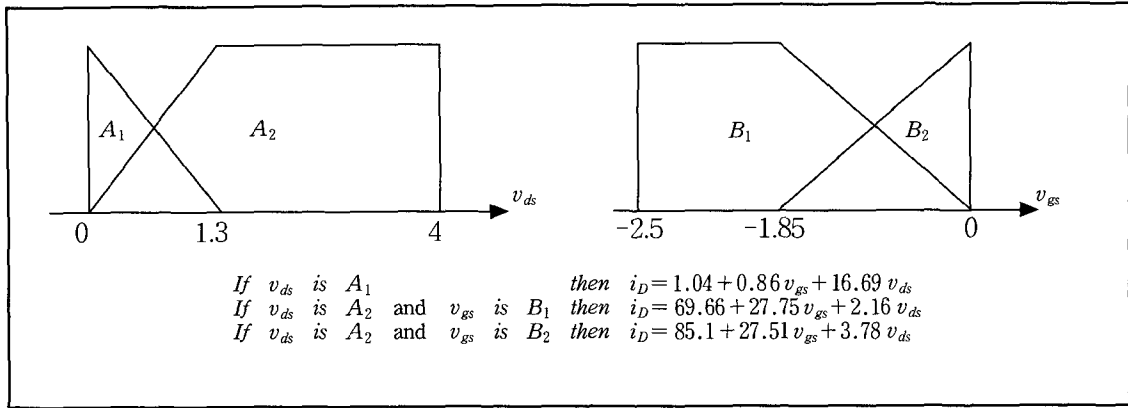


그림 4. GaAs FET의 TSK 퍼지모델
 Fig. 4. The TSK fuzzy model for GaAs FET.

은 많은 기억용량을 필요로 하고, 스플라인 함수는 대규모 회로의 계산에 비효율적이며, 구분선형 함수는 간단하지만 정확도가 매우 떨어진다. 따라서 본 예에서는 제안하는 기법을 반도체소자 특성의 퍼지모델링에 응용한다.

반도체 소자 중에서 물리적 해석에 의한 모델이 매우 복잡한 이온 주입식 GaAs FET의 전류 전압 특성 곡선의 모델링에 적용하였다. 모델의 출력은 드레인 전류이고 입력은 드레인 전압과 게이트 전압이다. Chua^[15]는 black box적인 모델링 수법의 하나로 piecewise-linear 모델을 제안하였으며, 강^[16]은 TSK 퍼지모델을 발표하였다. 그 모델들은 실험에서 얻은 입

출력 데이터를 이용하여 구해졌다. 문헌^[16]의 TSK 퍼지모델에서 전제부 적합도의 합이 1 이 되도록 ($\sum_{k=1}^n A^k(x) = 1$) 전제부 퍼지집합을 조금 수정한 TSK 퍼지모델을 그림 4에 보인다.

그림 5에는, 실제 측정된 데이터, 그림 3의 TSK 퍼지모델의 출력, 문헌^[10]의 piecewise-linear 모델의 출력들을 각각 보인다.

그림 5에서 보면, TSK 퍼지모델은 구분선형 모델보다 훨씬 나은 결과를 보이지만, 드레인 전압이 0인 경우 드레인 전류가 0이 되지 않고 있으며 낮은 드레인 전압과 낮은 게이트 전압에서 드레인 전류가 단조증가하지 않고 있다. 이러한 특정 부분의 수정은 TSK 퍼지

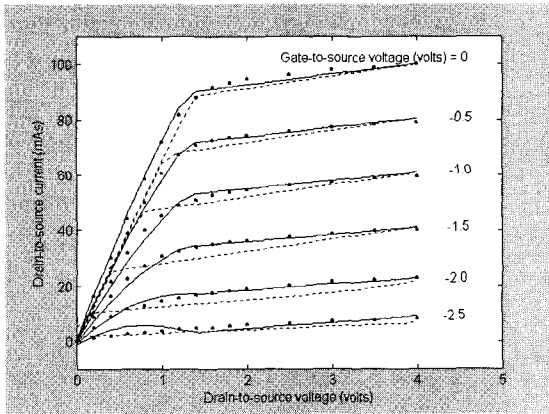


그림 5. GaAs FET 모델 출력 (원 : 측정된 데이터, 실선 : TSK 퍼지모델, 점선 : 구분선형모델)
 Fig. 5. The outputs of GaAs FET fuzzy model (circles : the measured data, solid lines : the TSK fuzzy model, dashed lines : piecewise liner model).

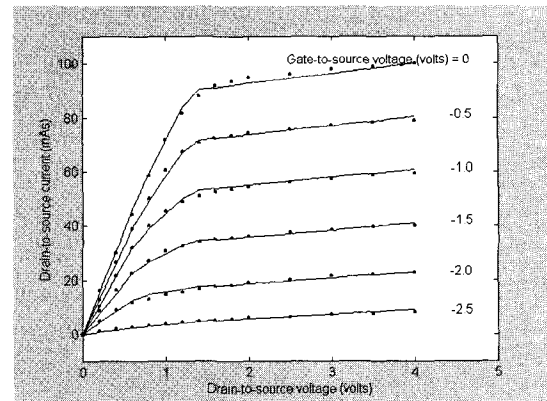


그림 7. GaAs FET 모델들의 출력 (원 : 측정된 데이터, 실선 : singleton 퍼지모델)
 Fig. 7. The outputs of GaAs FET models (circles : the measured data, solid lines : the singleton fuzzy model).

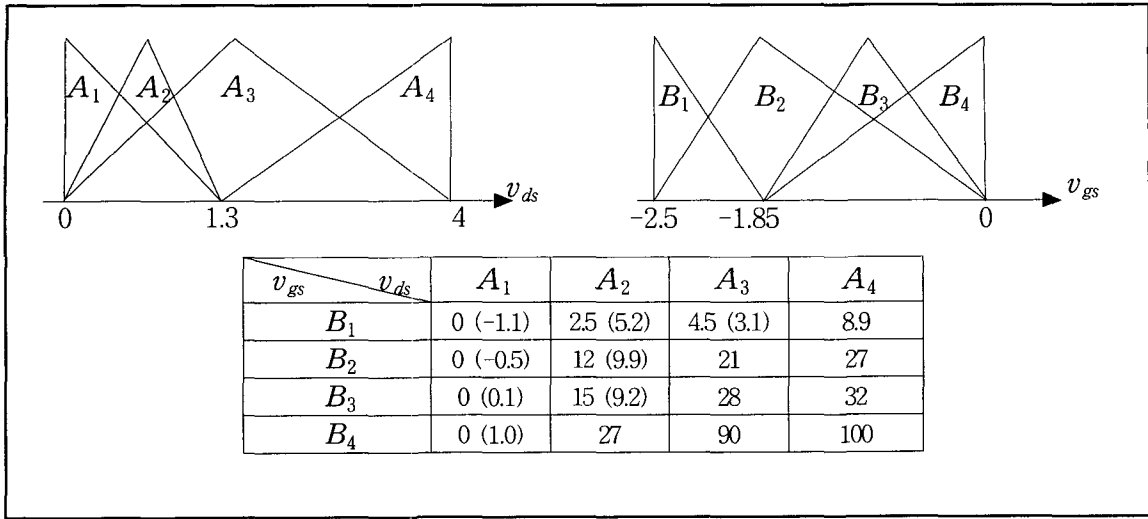


그림 6. GaAs FET의 singleton 퍼지모델
 Fig. 6. The singleton fuzzy model for GaAs FET.

모델의 형태로는 어렵다. 따라서 TSK 퍼지모델을 singleton 퍼지 모델로 변환시키고, 수정이 필요한 규칙의 결론부 실수 값을 수동으로 조정하였다. 그림 6에 조정된 singleton 퍼지모델을 보인다. 변환된 singleton 퍼지모델의 전제부 퍼지집합 멤버십 함수에서 곡선 부분은 계산의 간략화를 위해 직선 형태로 바꿨으며, 결론부 실수값에서 괄호 안의 숫자들은 조정하기 전의 값이다. 그림 7에 조정된 singleton 퍼지모델 출력을 보인다.

IV. 퍼지 제어기 설계에 응용

퍼지제어기의 설계 방법으로는, 제어 숙련가의 지식 등을 이용하여 trial-and-error적인 방법과 시스템 모델로부터 안정성을 생각하여 설계하는 방법으로 크게 나눌 수 있다. 결론부가 singleton 또는 퍼지집합인 퍼지제어기는 주로 제어 숙련가의 지식을 이용하여 설계된다. 그에 반해, TSK퍼지제어기의 경우, 결론부가 선형적인 장점을 이용하여 안정성을 고려하면서 퍼지모델로부터 체계적으로 다양한 형태로 설계될 수 있다.

본 장에서는 2장에서 제안한 변환 방법을 이용하면 singleton 퍼지제어기의 장점과 TSK 퍼지제어기의 장점을 모두 살릴 수 있음을 보여준다.

먼저 대상 시스템이 TSK퍼지모델로 표현되어 있을 때 그 퍼지모델로부터 극 배치법(pole placement)을 이용하여 TSK퍼지제어기를 설계한다. 이 때 설계된

TSK퍼지제어기는 TSK퍼지모델의 오차로 원하는 응답과 차이가 있을 수 있으며, 또는 제어기 사용자가 특정 상황에서 제어기 성능을 조정할 필요가 있을 수도 있다. 이러한 경우 TSK퍼지제어기의 형태로는 조정이 쉽지 않다. 그러나, 2장에서 설명한 방법으로 TSK퍼지제어기를 singleton퍼지제어기로 변환시키면 필요한 부분에서 제어기의 조정이 쉽게 가능하다.

지금부터, 극 배치법을 이용한 TSK퍼지제어기 설계법을 간단히 설명하고, TSK퍼지제어기를 singleton 퍼지제어기로 변환시키는 응용 예를 보인다.

1. 극 배치법을 이용한 TSK퍼지제어기 설계 [17,18]

TSK퍼지제어기를 설계하기 위해서는 먼저 시스템이 TSK 퍼지모델로 표현되어 있어야 한다. 시스템의 입출력 데이터로부터 TSK 퍼지모델이 인식되어 있을 경우 그 형태는 일반적으로 다음과 같다.

$$\begin{aligned}
 M^i: \text{ If } z_1(k) \text{ is } F_1^i, \quad z_2(k) \text{ is } F_2^i, \dots, z_n(k) \text{ is } F_n^i \\
 \text{ then } y^i(k+1) = a_0^i + a_1^i y(k) + a_2^i y(k-1) + \dots \\
 + a_n^i y(k-n+1) + b_1^i u(k) + b_2^i u(k-1) \\
 + \dots + b_m^i u(k-m+1)
 \end{aligned} \tag{21}$$

여기서 $z_i(k) = u(k), \forall k$ 라 가정한다. 위의 퍼지모델의 결론부를 상태변수표현으로 바꾸기 위하여

$$\begin{aligned}
 x_1(k) = y(k), \quad x_2(k) = y(k-1), \dots, \quad x_n(k) = y(k-n+1), \\
 d_1^i(k) = a_0^i + b_2^i u(k-1) + \dots + b_m^i u(k-m+1)
 \end{aligned}$$

라 하면 식 (21)의 퍼지규칙은 다음과 같이 쓸 수 있다.

$$M^i: \text{ If } z_1(k) \text{ is } F_1^i, z_2(k) \text{ is } F_2^i, \dots, z_n(k) \text{ is } F_n^i \\ \text{ then } x^i(k+1) = A^i x(k) + b^i u(k) + d^i(k) \\ y(k+1) = cx(k+1) \quad (22)$$

여기서

$$A^i = \begin{pmatrix} a_1^i & a_2^i & \dots & \dots & a_n^i \\ 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 & 0 \end{pmatrix}$$

$$b^i = (b_1^i \ 0 \ 0 \ 0 \ 0)^T \\ d^i(k) = (d_1^i(k) \ 0 \ 0 \ 0 \ 0)^T \\ c = (1 \ 0 \ \dots \ 0)$$

이다. 식 (21)의 퍼지모델과 식 (22)의 퍼지모델이 서로 동일함은 쉽게 확인할 수 있다.

다음은 극 배치법(pole placement)을 이용하여 TSK 퍼지모델로부터 TSK 퍼지제어기를 설계하는 방법을 설명한다. 식 (22)의 규칙 M^i 로부터 설계되는 퍼지제어기의 i 번째 규칙은 다음과 같다.

$$C^i: \text{ If } z_1(k) \text{ is } F_1^i, z_2(k) \text{ is } F_2^i, \dots, z_n(k) \text{ is } F_n^i \\ \text{ then } u^i(k) = -g^i x(k) + g_0^i(k) \quad (23)$$

여기서 $g^i = (g_1^i \ g_2^i \ \dots \ g_n^i)$ 이며 $g_0^i(k)$ 는 스칼라이고 규칙 C^i 의 전제부는 규칙 M^i 의 전제부와 동일하다. g^i 와 $g_0^i(k)$ 는 다음 식들로 구해진다.

$$\phi = A^i - b^i g^i \quad (24) \\ g_0^i(k) = -\frac{d_1^i(k)}{b_1^i}$$

여기서 ϕ 는 원하는 상태전이 행렬이다. 제어입력 $u(k)$ 는 다음 식의 consistence condition에 의해 구해진다.

$$u(k) = \frac{\sum_{i=1}^r w^i(z) b_1^i u^i(k)}{\sum_{i=1}^r w^i(z) b_1^i} \quad (25)$$

다음 정리는 위의 TSK 퍼지제어기를 사용할 경우 폐루프 시스템의 상태전이 행렬을 원하는 안정한 것으로 만들 수 있음을 보여준다.

정리^[18]: 식 (22)의 규칙들로 구성되는 퍼지모델을 식 (23)의 규칙들로 구성되는 퍼지제어기를 사용하여 제어할 경우, 폐루프 시스템의 거동은 상태전이 행렬이 원하는 안정한 행렬 ϕ 인 선형 시스템의 거동과 동일하게

된다.

증명:

$$x(k+1) = \frac{\sum_{i=1}^r w^i(z)(A^i x(k) + b^i u(k) + d^i(k))}{\sum_{i=1}^r w^i(z)} \\ = \frac{(\sum_{i=1}^r w^i(z)(A^i x(k) + d^i(k)) + \sum_{i=1}^r w^i(z) b^i u(k))}{\sum_{i=1}^r w^i(z)} \\ = \sum_{i=1}^r w^i(z)(A^i - b^i g^i)x(k) / \sum_{i=1}^r w^i(z) \\ = \phi x(k) \quad (26)$$

2. TSK퍼지제어기의 singleton퍼지제어기로 변환

위에서 설계된 TSK퍼지제어기를 singleton퍼지제어기로 변환시킬 경우 주의해야 할 점이 있다. TSK퍼지제어기에서 제어입력 $u(k)$ 를 구할 때 사용되는 consistence condition 식(25)는 TSK퍼지모델에서 출력을 구할 때 사용되는 식(2)와 차이가 있다. 즉 식(25)에서는 전제부 적합도 w^i 에 모델에서 $u(k)$ 의 계수 b_1^i 가 곱해진다. 따라서 II장의 방법을 사용하여 singleton 퍼지제어기로 변환시킬 때, TSK퍼지제어기 결론부 선형식에 b_1^i 를 미리 곱하여 변환하면 된다. 식(23)의 퍼지규칙들로 구성된 TSK퍼지제어기가 다음과 같은 singleton 퍼지제어기 규칙들로 변환 되었다 하자

$$S^j: \text{ If } v_1(k) \text{ is } G_1^j, v_2(k) \text{ is } G_2^j, \dots, v_m(k) \text{ is } G_m^j \\ \text{ then } u^j(k) = s^j \quad (27)$$

그러면 위의 singleton퍼지제어기로부터 제어입력은 식 (25)에 의해 다음과 같이 구해진다.

$$u(k) = \frac{\sum_{j=1}^r w_s^j(v) s^j}{\sum_{j=1}^r w^j(z) b_1^j} \quad (28)$$

여기서 $w_s^j(v) = \prod_{i=1}^m G_i^j(v_i)$ 는 singleton퍼지제어기 전제부 적합도이며, r_s 는 singleton퍼지제어기에서 규칙의 개수이다.

응용 예로서 다음의 시스템에 대해 퍼지제어기를 설계하였다.

$$y(k+1) = \frac{y(k)y(k-1) + u(k)}{1 + y(k)^2} \quad (29)$$

먼저 TSK퍼지모델을 인식하기 위해 구간 $[-1, 1]$ 에

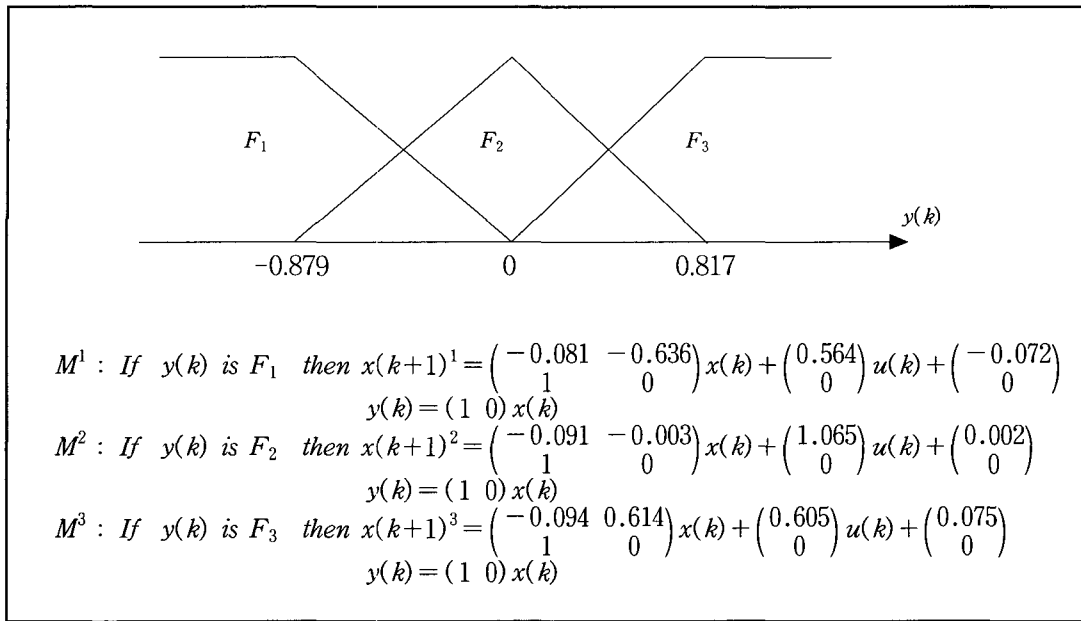


그림 8. 식 (29) 시스템의 TSK퍼지모델

Fig. 8. The TSK fuzzy model of system (29)

랜덤하게 분포한 $u(k)$ 를 식 (29)에 입력하여 1,000개의 입출력 데이터를 구하고, 그 데이터와 문헌^[5]의 알고리즘으로 TSK퍼지모델을 인식하여 결론부를 상태변수모델로 바꾼 결과 그림 8의 모델을 구하였다.

원하는 상태천이 행렬 ϕ 를

$$\phi = \begin{pmatrix} \phi_1 & \phi_2 \\ 1 & 0 \end{pmatrix}$$

라 하면 그림 8로부터 설계되는 TSK퍼지제어기는 다음과 같다.

$$\begin{aligned} C^1 : & \text{If } y(k) \text{ is } F_1 \text{ then } u(k)^1 \\ & = \begin{pmatrix} \phi_1 + 0.081 & \phi_2 + 0.636 \\ 0.564 & 0.564 \end{pmatrix} x(k) + \frac{0.072}{0.564} \\ C^2 : & \text{If } y(k) \text{ is } F_2 \text{ then } u(k)^2 \\ & = \begin{pmatrix} \phi_1 + 0.091 & \phi_2 + 0.003 \\ 1.065 & 1.065 \end{pmatrix} x(k) - \frac{0.002}{1.065} \\ C^3 : & \text{If } y(k) \text{ is } F_3 \text{ then } u(k)^3 \\ & = \begin{pmatrix} \phi_1 + 0.094 & \phi_2 - 0.614 \\ 0.605 & 0.605 \end{pmatrix} x(k) - \frac{0.075}{0.605} \end{aligned} \quad (30)$$

페루프 시스템 전달함수의 원하는 pole을 0.5 ($\phi_1 = 0.5, \phi_2 = -0.25$)로 하여 TSK퍼지제어기를 만들고, 4가지 초기값(-1, -0.5, 0.5, 1)에 대해 그 퍼지제어기를 적용하였을 때 응답을 그림 9에 보인다.

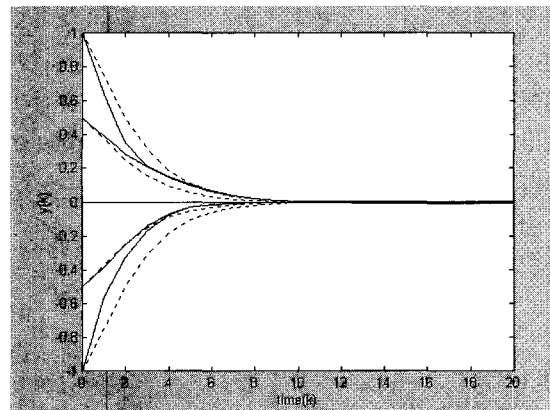


그림 9. 실선: TSK퍼지제어기의 제어결과, 점선: 원하는 응답

Fig. 9. Solid line: responses of TSK fuzzy controller, dashed line: desired responses.

그림 9에서 제어 결과 응답은 원하는 응답에 크게는 비슷하지만, 초기값에 따라 천이 응답 부분에서 원하는 응답과 조금 차이가 있는 부분이 있다. 이러한 제어 오차들은 식(29)의 원래 시스템과 그림 8의 TSK 퍼지모델의 차이에서 나온 것이며, TSK 제어기 형태로는 제어기 파라미터 조정으로 그 오차를 없애기가 어렵다. 하지만 singleton제어기 형태에서는 제어기 해석이 쉬우며 제어기 파라미터 조정이 간편하므로, 식 (30)의

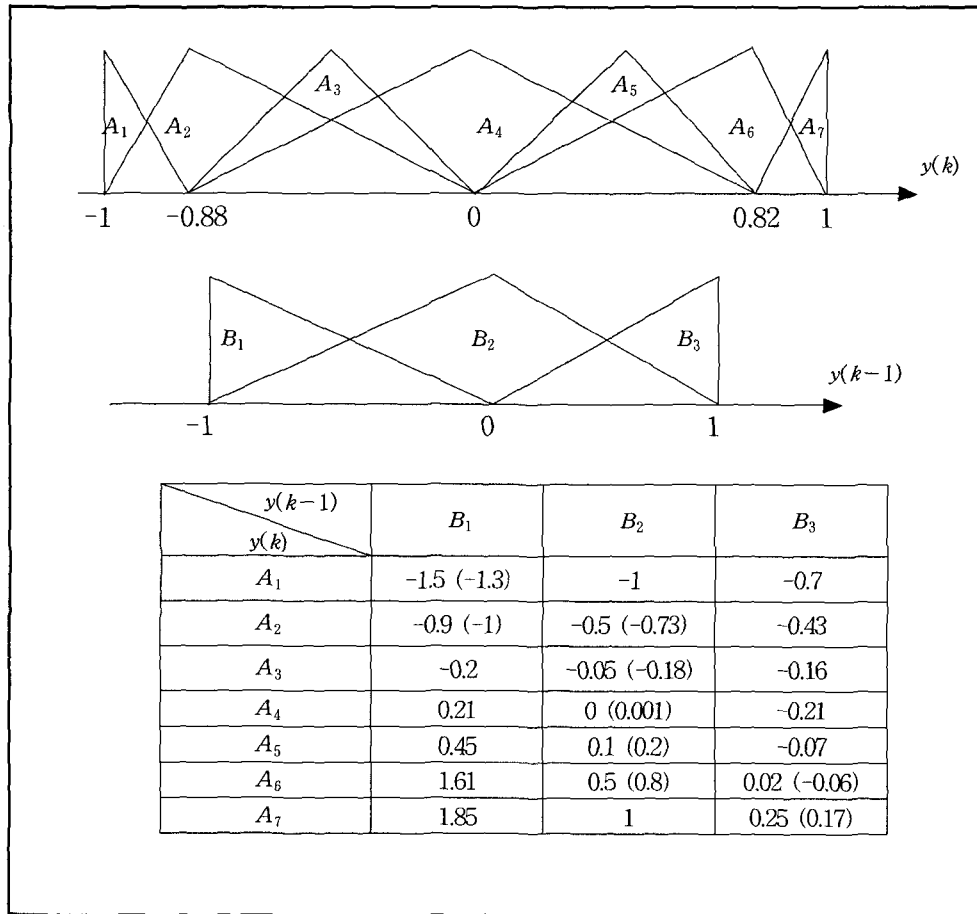


그림 10. 조정된 singleton 퍼지제어기
Fig. 10. The adjusted singleton fuzzy controller.

TSK퍼지제어기를 II장에서 설명한 방법으로 singleton 퍼지제어기로 변환시키고, 필요한 부분의 파라미터를 조정을 하였다. 그 방법은 조정이 필요한 공간에 해당하는 singleton 퍼지 규칙의 결론부 상수를 시행착오적인 방법으로 조정하였다. 그림 10에 singleton 퍼지제어기로 변환되고 조정된 결과를 보인다. 그림 10의 결론부 실수값에서 괄호안의 숫자는 조정하기 전의 파라미터 값이며, 전제부 퍼지집합 멤버쉽 함수에서 곡선 부분은 계산의 간략화를 위해 직선 형태로 하였다. 그리고, $y(k-1)$ 변수는 TSK퍼지제어기에서 결론부에서만 존재하므로 변환된 singleton 퍼지제어기에서는 2개의 퍼지집합으로 분할하면 충분하나, 좀 더 세밀한 조정을 위해 3개의 퍼지집합(B_1, B_2, B_3)으로 분할하였다.

그림 11에 조정된 singleton 퍼지제어기를 사용할 경우 응답을 보이며, 그 응답은 원하는 응답과 거의 일치함을 볼 수 있다. TSK퍼지제어기를 설계하여 그것을

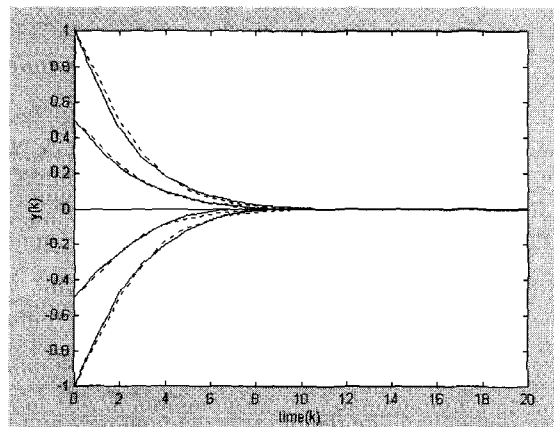


그림 11. 실선: 조정된 singleton 퍼지제어기의 응답, 점선: 원하는 응답
Fig. 11. Solid line: the responses of adjusted singleton fuzzy controller, dashed line: desired responses.

변환하지 않고 바로 그림 10과 같은 singleton 퍼지제어기를 설계하기는 매우 어렵다. 하지만 II장에서 설명한 변환 방법을 사용하면 TSK퍼지제어기와 singleton 퍼지제어기의 장점을 잘 살릴 수 있음을 알 수 있다.

V. 결 론

본 논문에서는 어느 한 TSK 퍼지시스템이 주어졌을 때 그 퍼지시스템과 동일한 입출력 관계를 갖는 singleton 퍼지시스템을 구하는 방법을 제안하고 응용 예를 보였다. TSK퍼지시스템의 전제부 퍼지집합 멤버십 함수가 사다리꼴을 할 경우 TSK퍼지시스템과 똑같은 입출력 관계를 갖는 singleton퍼지시스템을 쉽게 구할 수 있었다.

제안된 변환 방법을 이용하면 두 퍼지시스템의 장점을 잘 살릴 수 있음을 2개의 예로서 보였다.

먼저 퍼지 모델링의 예에서는, 입출력 데이터로 인식된 TSK퍼지모델을 singleton퍼지모델로 변환 시켜 원하는 부분을 조정하였다. 이 예를 통해, 입출력 데이터로 퍼지모델링 할 때, TSK퍼지모델에서 규칙 수가 적어 모델 구조 인식이 비교적 쉬운 장점과 singleton퍼지모델에서 모델 해석과 미세 조정이 쉬운 장점을 잘 이용할 수 있음을 보였다.

퍼지제어기 설계 예에서는, TSK퍼지모델로부터 극 배치법(pole placement)을 이용하여 TSK퍼지제어기를 설계하고 그 제어기를 singleton퍼지제어기로 변환시켜 필요한 부분을 조정하였다. 이 예로서 TSK퍼지모델의 경우 결론부가 선형적인 점을 이용하여 안정성을 보장하는 퍼지제어기 설계가 용이한 장점과 이해하기 쉽고 미세 조정이 용이한 singleton퍼지제어기의 장점을 모두 살릴 수 있음을 보였다.

참 고 문 헌

- [1] L. Zadeh, "Outline of a new approach to the analysis of complex systems and decision processes", *IEEE Trans. on SMC*, 3, pp. 28-44, 1973.
- [2] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modelling and control", *IEEE Trans. on SMC*, 15, pp. 116-132, 1985.
- [3] J. Buckley, "Sugeno type controllers are universal controller", *Fuzzy Sets and Systems*, vol.53, pp.199-303, 1993.
- [4] K. Zeng, N. Zhang, and W. Xu, "A Comparative study on sufficient conditions for Takagi-Sugeno fuzzy systems as universal approximators", *IEEE Trans. on Fuzzy Systems*, 8, pp. 773-780, 2000.
- [5] M. Sugeno and G. Kang, "Structure identification of fuzzy model", *Fuzzy Sets and Systems*, 28, pp.15-33, 1988.
- [6] L. Wang and R. Langari, "building Sugeno type models using fuzzy discretization and orthogonal parameter estimation techniques", *IEEE Trans. on Fuzzy Systems*, 3, pp. 454-458, 1995.
- [7] L. Wang and R. Langari, "Complex systems modeling via fuzzy logic", *IEEE Trans. on SMC-B*, 26, pp. 100-106, 1996.
- [8] E. Kim, M. Park, S. Ji and M. Park, "A new approach to fuzzy modeling", *IEEE Trans. on Fuzzy Systems*, 5, pp. 328-337, 1997.
- [9] K. Tanaka and M. Sugeno, "Stability analysis and design of fuzzy control system", *Fuzzy Sets and Systems*, 45, pp.135-156, 1992.
- [10] G. Kang and W. Lee, "Design of Fuzzy State Controllers and Observers", *Proc. IEEE International Conference on Fuzzy Systems*, pp. 1355-1360, 1995.
- [11] H. Wang, K. Tanak, and M. Griffin, "an approach to fuzzy control of nonlinear systems: Stability and design issues", *IEEE Trans. on Fuzzy Systems*, 4, pp. 14-23, 1996.
- [12] H. Lam, F. Leung, and P. Tam, "Stable and robust fuzzy control for uncertain nonlinear systems", *IEEE Trans. on SMC-A*, 30, pp. 825-840, 2000.
- [13] R. Babuska and H. Verbruggen, "A new identification method for linguistic fuzzy models", *Proc. 1995 IEEE International Conference on Fuzzy Systems*, Vol. 2. pp. 905

- 912, 1995.
- [14] Y. Chae, K. Oh, W. Lee, G. Kang, "Transformation of TSK fuzzy system into fuzzy system with singleton consequents and its application" *Proc. 1999. FUZZ-IEEE '99*. Vol. 2, pp. 969 - 973, 1999.
- [15] L. Chua and A. Deng, "Canonical piecewise linear modelling", *IEEE Trans. on Circuits Syst., CAS-33*, pp. 511-525, 1986.
- [16] 강근택, 권태하, "반도체소자의 퍼지모델", *대한전자공학회 논문집* 26-12, pp. 106-114, 1986.
- [17] G. Kang and W. Lee, "Design of Fuzzy State Controllers and Observers", *Proc. IEEE International Conference on Fuzzy Systems*, pp. 1355-1360, 1995.
- [18] 강근택, 이원창, "TSK퍼지모델로부터 TSK퍼지 제어기의 설계", *대한전자공학회 논문집 S편*, 35-11, pp. 1515-1529, 1998.

저 자 소 개



蔡 良 範(正會員)

1982년 2월 한국해양대학교 항해학과(공학사). 1984년 2월 한국해양대학교 대학원 항해학과(공학석사). 1997년 2월 부경대학교 대학원 전자공학과 박사과정 수료. 1989년 5월~현재 한국해양대학교 운항시스템공학부 교수

템공학부 교수

李 遠 昌(正會員) 論文誌 第26卷 第12號 參照

현재 부경대학교 전자컴퓨터정보통신공학부 부교수

姜 根 澤(正會員) 論文誌 第26卷 第12號 參照

현재 부경대학교 전자컴퓨터정보통신공학부 교수