

퍼지 시스템의 2계층 퍼지 시스템으로의 변환 방법 A method of converting fuzzy system into 2 layered hierarchical fuzzy system

주문갑
Moon G. Joo

부경대학교 전자컴퓨터정보통신공학부

요약

본 논문에서는 다입력 퍼지 로직 시스템에서 생기는 퍼지 규칙수의 기하급수적 증가를 막기 위하여, 주어진 퍼지 시스템의 THEN 부분을 이용하여 퍼지 규칙 벡터를 정의하고, 이를 이용하는 2 계층의 계층 퍼지 시스템으로 변환하는 방법을 제시한다. 여기에서, 1번째 계층에서는 주어진 퍼지 시스템으로부터 생성되는 일차독립의 퍼지 규칙 벡터를 사용하고, 2계층에서는 1계층에서 사용된 퍼지 규칙 벡터들의 선형합을 사용한다. 변환된 2계층의 퍼지 시스템은 주어진 퍼지 시스템과 동일한 근사 능력을 가질 뿐 아니라, 더 적은 수의 퍼지 규칙을 가짐을 보인다.

키워드 : 계층 퍼지 시스템, 퍼지 규칙 벡터, 선형합

Abstract

To solve the rule explosion problem in multi input fuzzy logic system, a method of converting a given fuzzy system to 2 layered hierarchical fuzzy system is presented where the collection of the THEN-parts of the fuzzy rules of given fuzzy system is considered as vectors of fuzzy rule. At the 1st layer, linearly independent fuzzy rule vectors generated from the given fuzzy logic system are used and, at the 2nd layer, linear combinations of these independent fuzzy rule vectors are used for fuzzy logic units at each layer. The resultant 2 layered hierarchical fuzzy system has not only equivalent approximation capability, but less number of fuzzy rules compared with the conventional fuzzy logic system.

Key words : Hierarchical fuzzy system, fuzzy rule vector, linear combination

1. 서론

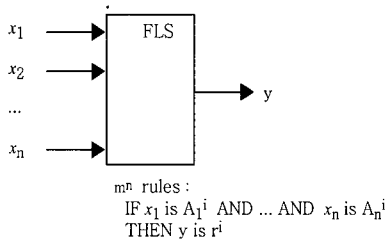


Figure 1: A conventional FLS.

퍼지 로직 시스템(Fuzzy Logic System, FLS)은 Zadeh가 그 이론을 발표한지 50년도 못되었지만, 여러 분야에서 괄목할 만한 연구 결과를 내고 있다. 현재

는 제어 시스템에 관련한 경우, 단순한 퍼지 시스템만의 응용을 넘어, 퍼지 시스템의 수학적 성질에 대한 연구[1]-[5]는 물론, sliding mode 제어 기법이나 Lyapunov 안정성 증명을 포함한 안정 적응 제어 기법[6][7]을 함께 이용하는 데까지 이르렀다. 한편, 퍼지 시스템에서 필요한 퍼지규칙의 수는 입력변수의 갯수에 따라 기하급수적으로 증가한다. 그림 1과 같은 전형적인 퍼지 로직 시스템의 경우를 예를 들어보자. 기존의 퍼지 로직 시스템에서 입력 변수의 개수가 4인 경우(n=4), 다음과 같은 형태의 퍼지 규칙이 필요하다.

IF x_1 is A_1 and x_2 is A_2 and x_3 is A_3 and x_4 is A_4
THEN y is r .

여기에서 x_1, x_2, x_3, x_4 는 입력변수, A_1, A_2, A_3, A_4 는 각각의 입력변수에 따른 퍼지 집합, y 는 출력변수, r 은 실수로 나타내어진다. 만약 각각의 입력 변수에 예를

접수일자 : 2006년 3월 30일

완료일자 : 2006년 6월 7일

들어 (크다, 조금 크다, 정확하다, 조금 작다, 작다)라는 5가지의 의미를 부여하게 되면($m=5$), 모든 경우를 고려한 퍼지 규칙의 수는 $m^n = 5^4 = 625$ 개가 된다. 같은 경우에 $n=6$ 의 경우에는 퍼지 규칙의 수는 $5^6 = 3125$ 개가 필요하며, 이를 통해 퍼지 규칙의 수가 입력 변수의 수에 따라 기하급수적으로 증가하게 됨을 알 수 있다. 시스템에서 사용되는 입력 변수의 개수에 따라 퍼지 규칙의 수가 기하급수적으로 증가한다는 사실은, 결국 메모리나 수행 속도, 퍼지 규칙 설정의 어려움 등으로 인해 적용상의 제약을 가져오므로, 가능한 퍼지 규칙의 수를 줄이는 것이 반드시 필요하다. 퍼지 로직 시스템에서 기하급수적으로 늘어나는 퍼지 규칙의 갯수를 줄이기 위하여 여러 형태의 분할 방안 및 계층화 [8]-[14]가 연구되었는데, 그 중 그림 2와 같은 형태의 계층 퍼지 시스템[15][16]이 제안되었다.

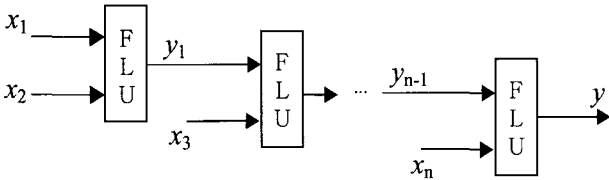


Figure 2: Hierarchical fuzzy system.

여기에서 k계층의 퍼지 로직 유닛(Fuzzy Logic Unit, FLU)은 다음과 같은 형태의 퍼지 규칙을 사용한다.

IF x_k is A and y_{k-1} is B THEN y_k is r,

즉, 각 계층에서는 퍼지 규칙을 설정하기 위하여, 입력 변수($x_1, x_2 \dots$)들뿐만 아니라 이전 계층의 출력(y_{k-1})을 함께 사용하여 현재 계층의 출력(y_k)을 결정하는 것이다. 이러한 구조의 계층 퍼지 시스템을 사용하면, 최소 퍼지 규칙의 수는 $(n-1)m^2$ 가 되는데, $n=4, m=5$ 인 경우는 $(4-1)*(5*5)=75$ 개가 되어, 기존 퍼지 로직 시스템의 625개에 비하여 크게 작다. 더 나아가 $n=6, m=5$ 인 경우는 $(6-1)*(5*5)=125$ 개로, 기존 퍼지 시스템이 3125개의 퍼지 규칙을 가지는 것에 비하여 현저히 작은 수의 퍼지 규칙을 가지게 된다. 그러나, 이러한 형태의 계층 퍼지 시스템이 나타낼 수 있는 함수의 종류는 제한되어 있기 때문에 이 형태의 계층 퍼지 시스템을 응용하는 경우, 적용하기에 적합한 대상을 골라 개발자의 직관에 따라 구현한 것이 대부분이다. 또한 기존의 퍼지 로직 시스템을 계층 퍼지 시스템으로 변환하는 일관된 방법이 존재하지 않기 때문에, 시스템의 입력 변수 중 중요도가 높은 입력 변수를 첫 번째 계층에 배치하고 그 다음 중요한 입력 변수를 두 번째 계층에 배치하는 방법, 혹은 정반대로 결정적인 역할을 하는 입력 변수일 수록 마지막 계층에 배치하는 방법을 쓰는 등, 개발자가 직관에 의하여 계층 퍼지 시스템을 구현해왔다. 계층 퍼지 시스템은 퍼지 규칙의 개수를 획기적으로 줄이는 데에는 성공하였으나, 계층이 많아질수록 각 계층에서 입력으로 사용되는 이전 계층의 출력(y_{k-1})에 대한 물리적인 의미를 부여하기가 점점 어려워지는 단점도 가지고 있다. 만약, 각 계층의 출력에

대하여 물리적인 의미를 계속해서 부여하지 못하면 사용자에게 의미있는 퍼지 규칙을 만들어 낼 수 없게 되고, 결과적으로 퍼지 시스템으로서의 장점을 잃어버리게 된다. 또한 계층 퍼지 시스템에서는 각 계층의 퍼지 로직 유닛들의 입력 변수의 선택과 계층 구조의 선택 방법도 문제가 된다. 이 문제를 해결하기 위한 방법으로 유전자 알고리즘과 Back propagation을 사용하는 경우[17][18]가 제안되었지만, 이런 방법으로 자동으로 구성된 계층 퍼지 시스템은 생성된 퍼지 규칙을 사용자가 이해하기가 거의 불가능해서, 퍼지 시스템의 장점인 퍼지 규칙의 가독성을 잃어버리게 된다. 기술한 계층 퍼지 시스템의 단점을 극복하기 위하여, 각 계층마다 명확한 의미를 가진 원래의 시스템 입력란을 입력 변수로 사용하는 계층 퍼지 시스템[4][5]이 제안되었고, 다음과 같은 형태의 퍼지 규칙이 사용되었다.

IF x_1 is A_1 AND x_2 is A_2 AND x_3 is $A_3 \dots$
 THEN y_k is $ay_{(k-1)1} + by_{(k-1)2} + cy_{(k-1)1}y_{(k-1)2} + d$

즉, 이전 계층의 출력값들을 퍼지 규칙의 전건부에서는 제외시키고, 대신 출력부에서 이전 계층의 출력들을 가중치를 주어 받아들이는 방법을 썼다. 이 형태의 계층 퍼지 시스템에 있어서, 퍼지 규칙의 수는 여러 응용들에서 기존의 퍼지 시스템에 비하여 작게 만들 수 있음을 보였지만, 기존의 퍼지 시스템에 비하여 반드시 작아진다고 말할 수는 없다. 다시 말해, 기존의 계층 퍼지 시스템이 퍼지 규칙의 수를 줄이는 데에 성공한 반면 가독성을 잃어버리는 경향이 있고, 이것을 극복하기 위해 제안된 새로운 형태의 계층 퍼지 시스템은 가독성이 향상된 반면 퍼지 규칙의 개수를 줄이는 데에는 약점이 있는 것이다. 본 논문에서는, 다입력 퍼지 로직 시스템에서 생기는 퍼지 규칙수의 기하급수적 증가를 막기 위하여 새로운 2계층의 퍼지 시스템을 구성하는 방법을 제시한다. 주어진 퍼지 시스템의 출력부 값을 이용하여 퍼지 규칙 벡터를 정의하고, 이를 이용하는 2계층의 계층 퍼지 시스템으로 변환한다. 변환된 2계층의 퍼지 시스템은 주어진 퍼지 시스템과 동일한 근사 능력을 가질 뿐 아니라, 더 적은 수의 퍼지 규칙을 가짐을 보인다.

2. 퍼지 시스템의 변환 방법

2.1 2계층 퍼지 시스템

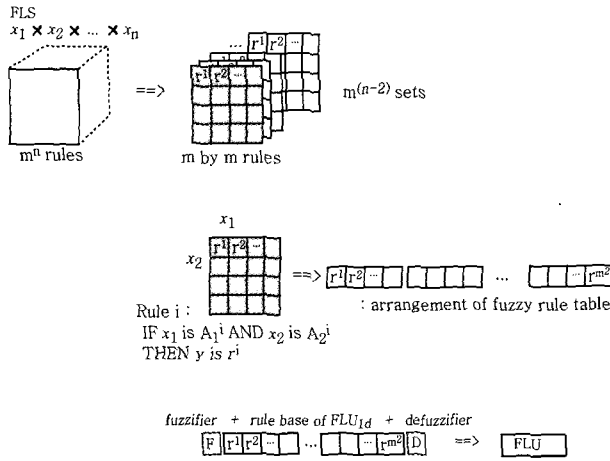


Figure 3: Arrangement of fuzzy rule table.

입력 변수의 갯수가 n 인 그림 1과 퍼지 로직 시스템(FLS)의 퍼지 규칙의 후건부들을 모아 그림 3의 상부 좌측에 보여지는 것처럼 THEN 부분의 출력값들로 이루어진 n 차원의 하이퍼큐브를 생각한다. n 차원의 하이퍼큐브는 그림의 상부 우측에서 보여지는 바와 같이 다수의 2차원 테이블로 분할할 수 있는데, 2차원 테이블은 2개의 입력 x_1, x_2 를 가진 퍼지 시스템의 퍼지 규칙에 해당한다. 또한 2차원 퍼지 규칙은 그림의 중간에서 보여지는 것처럼 늘어뜨린 1차원의 퍼지 규칙으로 나타낼 수 있다. 다시 말해, 기존의 n 차원 퍼지 규칙은 $m^{(n-2)}$ 개의, m^2 개 원소를 가진 1차원 퍼지 규칙 후건부값들의 모음으로 나타낼 수 있다. 이렇게 만들어진 1차원의 퍼지 규칙과 퍼지화기(fuzzifier) 및 비퍼지화기(defuzzifier)를 사용하여 그림 3의 아래에 나타낸 것과 같은 FLU_{1d} 를 정의한다. 여기에서 아래첨자는 1번째 계층의 d 번째 퍼지 로직 유닛(Fuzzy Logic Unit, FLU)라는 것을 나타낸다. 싱글톤 퍼지화기와 아래의 중심-평균 비퍼지화기(center-average defuzzifier)를 사용한다.

$$y = \frac{\sum_i \mu^i r^i}{\sum_i \mu^i}$$

여기에서 μ^i 는 i 번째 퍼지 규칙의 적합도를 나타낸다.

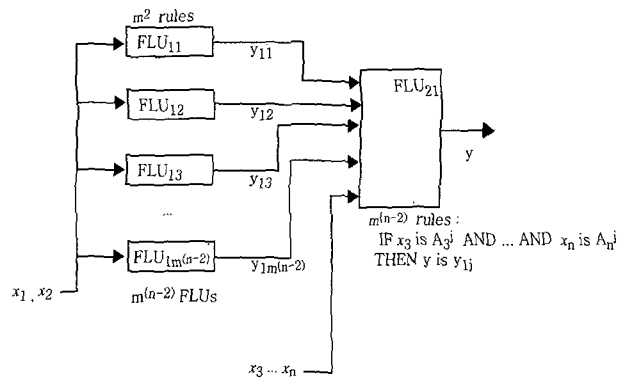


Figure 4: An equivalent HFS to conventional FLS.

그림 4는 상기 내용을 이용하여, 기존의 퍼지 시스템을 2계층의 퍼지 시스템으로 구성한 것이다. 시스템의 입력 변수 중, x_1, x_2 를 사용한 FLU들을 1계층에 배치하고, 2계층에서는 사용되지 않은 나머지 입력 변수들을 이용하여 1계층의 FLU의 출력값들을 선택하는 구조로 구성된다. 다시말해 1계층에서 x_1, x_2 를 이용하여 생성되는 출력값들을 2계층의 x_3, x_4, \dots, x_n 로써 적절히 선택하는 구조를 가진다. 예를 들어 5개의 시스템 입력 변수와 3개의 멤버십 함수를 가진 퍼지 시스템을 아래 구조로 변환하게 되면 1계층에 2개의 입력 변수를 가진 $3^{(5-2)} = 27$ 개의 FLU들이 사용되고, 2계층에서 3개의 입력 변수를 가진 1개의 FLU가 사용되는 형태가 된다. 그림 4의 HFS에서 1번째 계층의 d 번째 FLU_{1d} 의 i 번째 퍼지 규칙은

$$\text{IF } x_1 \text{ is } A_1^i \text{ and } x_2 \text{ is } A_2^i \text{ THEN } y_{1d} \text{ is } r_{1d}^i$$

로 정의되고, 2번째 계층의 FLU의 j 번째 퍼지 규칙은

$$\text{IF } x_3 \text{ is } A_3^j \text{ AND } \dots \text{ AND } x_n \text{ is } A_n^j \text{ THEN } y \text{ is } y_{1j}$$

이며, $i \in \{1, 2, \dots, m^2\}$, $d \in \{1, 2, \dots, m^{(n-2)}\}$, $j \in \{1, 2, \dots, m^{(n-2)}\}$, y_{1d} 는 FLU_{1d} 의 출력, y 는 전체 HFS의 출력을 나타낸다.

2.2 2계층 퍼지 제어기의 동등성

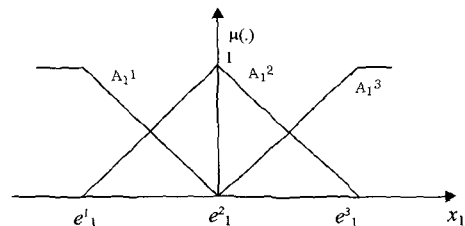


Figure 5: Center points.

그림 5와 같은 삼각 함수를 입력 멤버십 함수로 사용하고, $e_1^1, e_1^2, e_1^3, \dots$ 를 각각 x_1, x_2, x_3, \dots 의 멤버십 함수의 값이 1이 되는 값들이라고 정의하자. 이

때, 2계층의 퍼지 시스템의 입력 변수 (x_3, x_4, \dots)의 값이 (e_3^K, e_4^L, \dots)인 모든 지점에서는 2번째 계층의 FLU에서 단 한개의 퍼지 규칙의 적합도만이 1이 되고, 나머지는 0이 되어, 하나의 퍼지 규칙만이 선택된다. 선택된 1계층 퍼지 시스템의 출력값은 (x_3, x_4, \dots) = (e_3^K, e_4^L, \dots)일 때의 x_1, x_2 를 사용하는 2차원 퍼지 규칙에 의하여 계산된, 입력이 ($x_1, x_2, e_3^K, e_4^L, \dots$)의 결과를 말한다. 그러므로, 입력이 ($e_1^I, e_2^J, e_3^K, e_4^L, \dots$)인 모든 격자점에서, 제안된 2계층 퍼지 제어기의 출력은 기존의 퍼지 제어기의 출력과 같으며, 이는 기존의 퍼지 제어기를 2계층의 퍼지 제어기로 동등하게 변환할 수 있음을 의미한다.

$$HFS(e_1^I, e_2^J, e_3^K, e_4^L, \dots) = FLS(e_1^I, e_2^J, e_3^K, e_4^L, \dots)$$

2.3 2계층 퍼지 시스템의 최적화

위와 같은 2계층 퍼지 시스템에서 1번째 계층의 퍼지 시스템의 개수를 줄이기 위하여, 퍼지 규칙 벡터를 정의하고 선형 대수를 사용한다. 기존 퍼지 시스템에서 주어지는 n 차원의 퍼지 규칙을 그림 3에서 보여지는 것처럼 1차원의 퍼지 규칙 벡터로 분할하고, 분할된 퍼지 규칙 벡터 중 서로 선형 독립 성질을 가지는 것들만 골라 1계층의 FLU들을 구성하고, 2계층의 퍼지 시스템은 1계층 FLU 출력값의 선형합을 구성하는 퍼지 규칙을 가지도록 구성한다. 먼저 1계층의 퍼지 규칙들로부터 다음과 같은 퍼지 규칙 행렬을 구성하자.

$$R_B = \begin{bmatrix} \text{FLU}_{11} \text{의 퍼지 규칙 벡터} \\ \text{FLU}_{12} \text{의 퍼지 규칙 벡터} \\ \dots \\ \text{FLU}_{1m^{(n-2)}} \text{의 퍼지 규칙 벡터} \end{bmatrix} = \begin{bmatrix} r_{11}^1 & r_{11}^2 & \dots & r_{11}^{m^2} \\ r_{12}^1 & r_{12}^2 & \dots & r_{12}^{m^2} \\ \vdots & \vdots & \ddots & \vdots \\ r_{1m^{(n-2)}}^1 & r_{1m^{(n-2)}}^2 & \dots & r_{1m^{(n-2)}}^{m^2} \end{bmatrix}$$

여기에서 아래 첨자는 1번째 계층의 FLU를 지정하고, 위 첨자는 지정된 FLU 내에서의 몇번째 퍼지 규칙인가를 표시한다. 한 행의 퍼지 규칙들은 하나의 FLU의 후건부 값들의 모음으로서, 퍼지 규칙 벡터라고 정의한다. 만약 입력 변수의 갯수가 $n \geq 4$ 를 만족한다면 위 행렬 R_B 는 최대 m^2 개의 선형 독립의 퍼지 규칙 벡터를 가지게 된다. 이러한 선형독립 퍼지 규칙을 가진 FLU들을 $FLU_{1(i)}$ 라고 표시하고, 그 출력값을 $y_{1(i)}$ 라고 표시하여 그림 6과 같은 2계층의 퍼지 시스템으로 재구성한다.

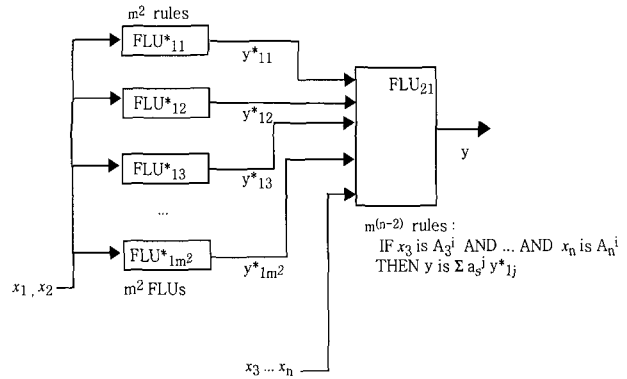


Figure 6: Optimal HFS.

여기에서 1번째 계층의 d 번째 FLU^* 의 i 번째 퍼지 규칙은

$$\text{IF } x_1 \text{ is } A_1^i \text{ AND } x_2 \text{ is } A_2^i \text{ THEN } y_{1d}^* \text{ is } r_{1d}^i,$$

이고, 2계층에서 j 번째 퍼지 규칙은

$$\text{IF } x_3 \text{ is } A_3^j \text{ AND } \dots \text{ AND } x_n \text{ is } A_n^j \text{ THEN } y \text{ is } \sum_{s=1}^{m^2} a_s^j y_{1s}^*$$

이며, $d \in \{1, 2, \dots, m^2\}$, $i \in \{1, 2, \dots, m^2\}$, $j \in \{1, 2, \dots, m^{n-2}\}$, a_s^j 는 나머지 퍼지 규칙 벡터들을 선형 독립 퍼지 규칙 벡터들의 선형합으로서 나타내기 위한 실계수로서 $n - 2 \geq 2$ 이면 항상 존재한다. 2계층의 퍼지 시스템의 퍼지 규칙 중 m^2 개는 1계층의 선형 독립 퍼지 규칙 중 하나를 선택하는 $(a_1, \dots, a_{m^2}) = (0, \dots, a_k = 1, \dots, 0)$, $k \in \{1, 2, \dots, m^2\}$ 의 형태로 미리 정해진다. 재구성된 최적화된 2계층 퍼지 시스템의 출력은 원래의 2계층 퍼지 시스템의 출력값과 같으므로, 이전과 같은 논리를 통하여, $HFS(e_1^I, e_2^J, e_3^K, e_4^L, \dots) = FLS(e_1^I, e_2^J, e_3^K, e_4^L, \dots)$ 임을 알 수 있다. 예를 들어, 5개의 시스템 입력 변수와 3개의 멤버십 함수를 가진 퍼지 시스템을 최적화된 2계층 퍼지 시스템으로 변환하는 경우, 1계층에는 x_1, x_2 의 입력 변수를 가진 최대 $3^2 = 9$ 개의 선형 독립 FLU들로 구성하고, 2계층에는 x_3, x_4, x_5 의 입력 변수를 가진 1개의 FLU로 구성한다. 이때 1계층 FLU는 각각 9개의 퍼지 규칙을 가지고, 2계층 FLU는 27개의 퍼지 규칙을 가지게 되어, 총 퍼지 규칙의 수는 $81 + 27 = 108$ 개가 된다.

2.4 제안된 2계층 퍼지 시스템의 일반화

상기 설명에서는 1계층 퍼지 시스템에 2개의 입력 변수를 사용하는 것을 기준으로 하였으나, 1계층 퍼지 시스템에 몇 개의 입력 변수를 사용하든 기본 변환 과정은 변하지 않는다. n 개의 입력 변수를 가진 기존의 퍼지 시스템을 제안된 2계층의 퍼지 시스템으로 변환하는 과정을 생각해 보자. p 개의 입력 변수(x_1, x_2, \dots, x_p)가 1번째 계층에서 사용되고, 나머지 $(n - p)$ 개 입력 변수($x_{p+1}, x_{p+2}, \dots, x_n$)는 2계층에서 사용된다고 가정한다.

(퍼지 규칙 벡터) 퍼지 규칙 벡터는 원래의 n 개 입력 변수를 가진 퍼지 시스템을 제안된 방법으로 p 개의 입력 변수를 가진 $m^{(n-p)}$ 개의 작은 퍼지 시스템으로 분할한 후, 분할된 퍼지 시스템의 후건부 값들로 구성된 $1 \times m^p$ 행렬을 말한다.

(퍼지 규칙 행렬) 퍼지 규칙 행렬은 주어진 퍼지 시스템의 모든 퍼지 규칙 벡터들로 이루어진 아래와 같은 $m^{(n-p)} \times m^p$ 행렬을 말한다.

$$R_B = \begin{bmatrix} r_{11}^1 & r_{11}^2 & \dots & r_{11}^{m^p} \\ r_{12}^1 & r_{12}^2 & \dots & r_{12}^{m^p} \\ \vdots & \vdots & \ddots & \vdots \\ r_{1m^{(n-p)}}^1 & r_{1m^{(n-p)}}^2 & \dots & r_{1m^{(n-p)}}^{m^p} \end{bmatrix}$$

만약 $n - p \geq p$ 를 만족하면, 최대 m^p 개의 선형 독립인 퍼지 규칙 벡터와 $m^{(n-p)} - m^p$ 개의 나머지 퍼지 규칙 벡터로 나눌 수 있다.

(2계층 퍼지 시스템으로의 변환) 제안된 2계층 퍼지 시스템의 최적 구성은 다음과 같다. 1번째 계층에서 d 번째 FLU*의 i 번째 퍼지 규칙은

$$\text{IF } x_1 \text{ is } A_1^i \text{ AND } \dots \text{ AND } x_p \text{ is } A_p^i \text{ THEN } y_{1d}^* \text{ is } r_{1d}^i,$$

이고, 2계층의 j 번째 퍼지 규칙은

$$\text{IF } x_{p+1} \text{ is } A_{p+1}^j \text{ AND } \dots \text{ AND } x_n \text{ is } A_n^j \\ \text{ THEN } y_{is} \text{ is } \sum_{s=1}^{m^p} a_s^j y_{1s}^*,$$

이며, $d \in \{1, 2, \dots, m^p\}$, $i \in \{1, 2, \dots, m^p\}$, $j \in \{1, 2, \dots, m^{n-p}\}$, a_s^j 는 선형 독립 퍼지 규칙 벡터들로 나머지 퍼지 규칙 벡터들을 선형합으로 나타낼 때 사용된 실수 계수들을 나타낸다.

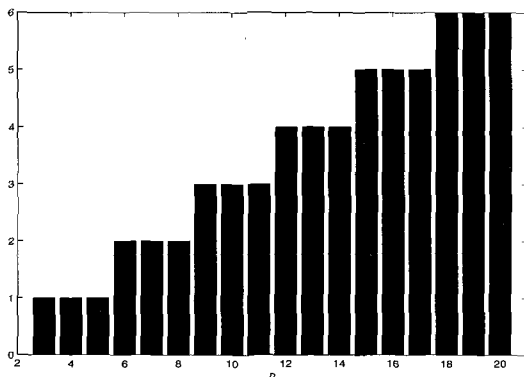


Figure 7: Optimal number of input variables at the 1st layer.

(퍼지 규칙의 갯수) 사용된 퍼지 규칙의 수는

$$m^p m^p + m^{n-p}.$$

이며, 퍼지 규칙의 수를 최소화할 수 있는 1계층 입력 변수의 갯수 p 를 그림 7에 나타내었는데, 계산 결과 m 의 값은 최적 갯수 p 를 구하는데에 영향을 미치지 않음을 알 수 있었다.

Example 1 입력 변수의 갯수 $n = 4$, 멤버쉽 함수의 갯수 $m = 3$ 일 때, 그림 7로부터 1번째 계층에서는 $p = 2$ 개의 입력 변수가 사용된다. 이 때, 사용된 퍼지 규칙의 수는 $3^2 3^2 + 3^{4-2} = 89$ 개이고, 기존의 퍼지 시스템에서는 $3^4 = 81$ 개의 퍼지 규칙이 사용된다.

Example 2 입력 변수의 갯수 $n = 5$, 멤버쉽 함수의 갯수 $m = 3$ 일 때, 그림 7로부터 1번째 계층에서는 $p = 2$ 개의 입력 변수가 사용된다. 이 때, 사용된 퍼지 규칙의 수는 $3^2 3^3 + 3^{5-2} = 108$ 개이고, 기존의 퍼지 시스템에서는 $3^5 = 243$ 개의 퍼지 규칙이 사용된다.

Example 3 입력 변수의 갯수 $n = 5$, 멤버쉽 함수의 갯수 $m = 5$ 일 때, 그림 7로부터 1번째 계층에서는 $p = 2$ 개의 입력 변수가 사용된다. 이 때, 사용된 퍼지 규칙의 수는 $5^2 5^3 + 5^{5-2} = 750$ 개이고, 기존의 퍼지 시스템에서는 $5^5 = 3125$ 개의 퍼지 규칙이 사용된다.

3. 결론

본 논문에서는 다입력 퍼지 로직 시스템에서 생기는 퍼지 규칙수의 기하급수적 증가를 막기 위하여, 주어진 퍼지 시스템을 2계층의 퍼지 시스템으로 변환하는 방법을 제시하였다. 먼저 주어진 퍼지 시스템의 출력부 값을 이용하여 퍼지 규칙 벡터를 정의하였다. 1번째 계층의 FLU는 일차 독립의 퍼지 규칙 벡터만을 사용하고, 2번째 계층의 FLU는 1계층에서 사용된 퍼지 규칙 벡터들의 선형합을 사용하여 1계층의 FLU의 수를 줄였다. 변환된 2계층의 퍼지 시스템은 삼각형 멤버쉽 함수를 사용하는 경우에 모든 격자점에서 주어진 퍼지 시스템과 동일한 값을 가짐을 보였으며, 주어진 퍼지 시스템보다 더 적은 수의 퍼지 규칙을 가짐을 보였다.

Acknowledgement. 이 논문은 2003년도 부경대학교 기성회 학술연구비에 의하여 연구되었음

References

[1] Buckley, J. J., "Sugeno type controllers are universal controllers," *Fuzzy Sets and Systems*, vol. 53, 1993, pp. 299-303.

[2] Huwendiek, O. and W. Brockmann, "Function approximation with decomposed fuzzy systems," *Fuzzy sets and systems*, vol. 101, 1999, pp. 273-286.

- [3] Wang, Li-Xin, "Universal approximation by hierarchical fuzzy systems," *Fuzzy sets and systems*, vol. 93, 1998, pp. 223-230.
- [4] Moon G. Joo and Jin S. Lee, "Universal approximation by hierarchical fuzzy system with constraints on the fuzzy rule," *Fuzzy Sets and Systems*, vol. 130. no. 2, 2002, pp. 175-188.
- [5] Moon G. Joo and Jin S. Lee, "A class of hierarchical fuzzy system with constraints on the fuzzy rule," *IEEE trans. Fuzzy System*, vol. 13. no. 2, 2005, pp. 194-203.
- [6] Ordonez, R. and Passino, K. M. "Stable multi-input multi-output adaptive fuzzy/neural control," *IEEE trans. Fuzzy System* vol. 30, no. 7, 1999, pp. 345-353.
- [7] M. G. Joo, Y. H. Kim, and T. Kang, "Stabel adaptive fuzzy control of molten steel level in the strip casting process," *IEE proceedings-Control Theory and Applications* vol. 149, no. 5, 2002, pp. 357-364.
- [8] Gegov, Alexander E., "Multilayer fuzzy control of multivariable systems by direct decomposition," *Int. Journal of systems science*, vol. 29, no. 8, 1998, pp. 851-862.
- [9] Gupta. Madan M., Jerzy B. Kiszka, and G. M. Trojan, "Multivariable structure of fuzzy control systems," *IEEE trans. on systems, man, and cybernetics*, vol. SMC-16, no. 5, Sep./Oct. 1986, pp. 638-655.
- [10] Lee, Pyeong G., Kyun K. Lee, and Gi J. Jeon, "An index of applicability for the decomposition method of multivariable fuzzy systems," *IEEE trans. on fuzzy systems*, vol. 3, no. 3, Aug. 1995, pp. 364-369.
- [11] Bolognani, Silverio and Mauro Zigliotto, "Hardware and software effective configurations for multi-input fuzzy logic controllers," *IEEE trans. on fuzzy systems*, vol. 6, No. 1, Feb. 1998, pp. 173-179.
- [12] Chen, Hung-Pin and Tai-Ming Parng, "A new approach of multi-stage fuzzy logic inference," *Fuzzy sets and systems*, vol. 78, 1996, pp. 51-72.
- [13] Chung, Fu-Lai and Ji-Cheng Duan, "On multistage fuzzy neural network modeling," *IEEE trans. on fuzzy systems*, vol. 8, no. 2, 2000, pp. 125-142.
- [14] Yager, Ronald R., "On the Construction of Hierarchical Fuzzy Systems Models," *IEEE trans. on systems, man, and cybernetics*, vol. 28, no. 1, Feb. 1998, pp. 55-66.
- [15] Raju, G. V. S., J. Zhou, and R. A. Kisner, "Hierarchical fuzzy control," *Int. J. Contr.*, vol. 54, no. 5, 1991, pp. 1201-1216.
- [16] Raju, G. V. S. and Jun Zhou, "Adaptive Hierarchical Fuzzy Controller," *IEEE trans. on systems, man and cybernetics*, vol. 23, no. 4, Jul. Aug. 1993, pp. 973-980.
- [17] Shimojima, Koji, Toshio Fukuda, and Yasuhisa Hasegawa, "Self-tuning fuzzy modeling with adaptive membership function, rules, and hierarchical structure based on genetic algorithm," *Fuzzy sets and systems*, vol. 71, 1995, pp. 295-309.
- [18] Linkens, Derek A. and H. Okola Nyongesa, "A hierarchical multivariable fuzzy controller for learning with genetic algorithms," *Int. J. Contr.*, vol. 63, no. 5, 1996, pp. 865-883.

저자 소개

주문갑

현재 부경대학교 전자컴퓨터정보통신공학부 조교수
 제 16권 1호(2006년 2월호) 참조
 E-mail : gabi@pknu.ac.kr