

# FUZZY CONTROL 이론 및 연구동향

姜 勳

中央大學校 制御計測工學科

## I. '퍼지(Fuzzy)'란 무엇인가?

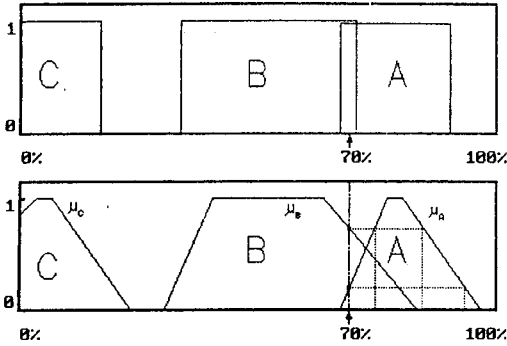
### 1. 퍼지논리와 퍼지집합

퍼지논리는 인공지능분야에서 논리의 공식화를 위해 1965년 Lotfi A. Zadeh 교수에 의해 최초로 시작되었으며,<sup>[1]</sup> 이는 '자동차 자동변속단 위치결정', '지하철 자동운행제어시스템', '에어컨/선풍기의 풍량 및 풍향조절', '냉장고 온도제어', '비행체 추진방향결정' 등 우리의 실제생활에 이미 적용되었거나 개발중이며, 물리적인 수치와 양에 대한 인간의 불확실한 논리적개념을 정연한 수학적 표현형태로 나타낼 수 있다는 특징이 있다. 또한, 퍼지논리는 공학뿐만 아니라 인간의 속성과 관련된 사회과학에의 논리적 접근에도 응용되고 있다.

부울(Boolean)논리를 기본으로 하는 기존의 집합이론에서는 특정한 오브젝트(Object)가 주어진 집합 A의 원소로서 속하거나(1,True) 속하지 않거나(0,False) 둘 중의 하나이다. 그러나 퍼지논리를 기반으로 하는 퍼지집합이론에서는 이 오브젝트가 구성원소로서 특정집합 A에 어느 정도로 속하는가 하는 것을 0(집합 A에 전혀 속하지 않음)에서부터 1(집합 A에 완전히 속함) 사이의 수치로써 나타내며 이를 소속의 정도(Degree of Membership)라고 한다. 이것이 퍼지논리가 인간의 불확실한 퍼지개념을 다룰 수 있도록 해준다. 예를 들어 부울논리는 "장미꽃(R)이 아름답다(A)"하는 주관적 명제를 'True' 또는 'False' 둘 중 오직 하나로만 나타내지만, 퍼지논리는 "R"이 "A"라는 집합 A(R)에 속하는 정도를 이룰데면  $\mu_{A(R)}=0.95$ 라는 수치를 부여하여 장미꽃이 아름다운 정도를 0과 1 사이의 숫자로 표현함으로써 일반 부울대수에서 표현하지 못하는 것을 나타낼 수 있다. 퍼지집합 A는 퍼지논리의 수학적 표현수단이며, 다음과 같이 정의된다<sup>[2]</sup> :

$$\mu_A(x) = \{(x, \mu_A) \mid x \in \text{실수}, 0 \leq \mu_A \leq 1\} \quad (1)$$

여기서,  $x$ 는 변수,  $\mu_A$ 는 소속함수(Membership Function)라고 한다. <그림 1>은 어떤 장미꽃의



〈그림 1〉 장미꽃의 아름다움에 대한 보통집합과 퍼지집합의 비교

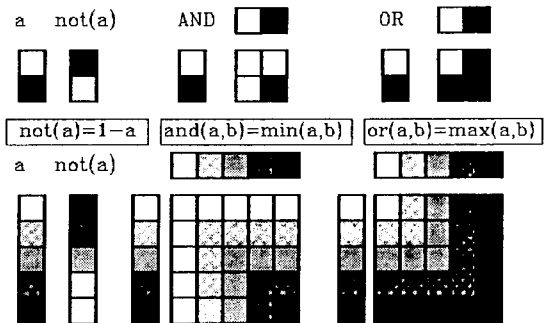
아름다움에 대한 측도를 0%에서 100%까지로 나타낸다고 할 경우 A=“아름답다”, B=“(아름다움이)그저 그렇다”, C=“아름답지 못하다” 등의 명제를 보통집합과 퍼지집합들을 가지고 비교한 것이다. 물론, 아름답다는 것은 주관적이고 사람마다 해석이 다를 수 있지만, 여기서는 감각적으로 비교될 수 있는 것으로 가정한다.

〈그림 1〉에서 보여주는 것처럼 퍼지집합은 하나의 측정값(70%)에 대해 소속의 정도가 A, B에 대해 각각 다른 값을 나타내지만 보통집합에 대해서는 그렇지 못하다. 반면에 소속의 정도가 1에 가까워지면 퍼지집합은 신뢰구간이 좁아지며, 0에 가까울수록 넓은 신뢰구간을 가지게 된다. 위의 예에서 집합 A와 B의 구별이 애매한 70%라는 값에 대해 보통집합에서는 실제로는 A인데 B가 되는 오류, 그리고 반대로 B인데 A로 잘못 판단될 경우의 문제점이 있다. 그러나, 퍼지집합에서는 A와 B모두를 0과 1 사이의 등급으로 표시함으로써 그러한 오류를 범할 가능성을 줄이는 효과가 있다. 이와 같이 퍼지집합이 보통집합과 다른 점은 A AND B라는 교집합, 즉, 중첩된 영역에 대해 서로 다른 소속의 정도를 갖는 외에도, 하나의 연속적 퍼지집합이 서로 다른 소속의 정도에 대하여 무한히 많은 보통집합의 합집합으로 구성되어 있다는 점이다. 이것을 ‘분해 등식(Resolution Identity)’이라고 한다. 퍼지논리는 부울논리와 마찬가지로 AND, OR, NOT 등과 같은 집합연산자에 대해서도

〈표 1〉 부울논리와 퍼지논리의 집합표준연산자의 비교

Boolean Set Operators	Fuzzy Set Operators (Standard)
$a, b \in \{0,1\}$	$a, b \in [0,1]$
AND( $a,b$ ) = 1 if $a=b=1$ = 0 otherwise	AND( $a,b$ ) = $\min(a,b)$
OR( $a,b$ ) = 0 if $a=b=0$ = 1 otherwise	OR( $a,b$ ) = $\max(a,b)$
NOT( $a$ ) = 0 if $a = 1$ = 1 if $a = 0$	NOT( $a$ ) = $1 - a$

**MULTI-VALUED LOGICS**



〈그림 2〉 퍼지논리를 Multi-valued Logic화 했을때 부울논리와와의 비교

퍼지논리 특유의 연산이 〈표 1〉과 같이 정의되어 있다. 그리고, 퍼지논리를 Multi-valued Logic으로 나타냈을 때의 부울논리와와의 비교가 〈그림 2〉에 나타나 있다.

**2. 퍼지이론 vs. 확률이론**

퍼지이론과 확률이론은 연관성이 조금은 있다고 볼 수 있다. 왜냐하면, 현대 확률이론의 바탕이 되는 Kolmogorov의 4가지 기본공리에서 마지막 네 번째 공리만이 서로 다르기 때문이다. 즉 확률에서는<sup>[3]</sup>

$$\begin{aligned} \text{Axiom 4(Probability)} : P(A \cup B) &= P(A) \\ &+ P(B) \text{ whenever } A \cap B = \emptyset \end{aligned} \quad (2)$$

이것은 확률의 “Inclusion & Exclusion 원리”에 의해, 일반적으로 다음과 같이 다시 쓸 수 있다 :

$$P(A \cap B) = P(A) + P(B) - P(A \cup B) \quad (3)$$

문이다.

반면, Plausibility 이론에서의 네번째 공리는

$$\begin{aligned} \text{Axiom 4(Plausibility)} : P(A \cap B) \leq P(A) \\ + P(B) - P(A \cup B) \end{aligned} \quad (4)$$

이고, 가능성이론에서의 네번째 공리는

$$\begin{aligned} \text{Axiom 4(Possibility)} : \pi(A \cup B) = \\ \max\{\pi(A), \pi(B)\} \end{aligned} \quad (5)$$

이다.<sup>[4]</sup> 퍼지이론은 가능성이론에 기초를 두고 있으며, 소속함수에 대해  $\mu_A(x) = \pi_x(A)$ 의 관계식이 정의된다. 이는 Plausibility 이론에서 연속적인 하나의 퍼지집합을 “분해 등식”을 사용하여 무한한 보통집합으로 분해했을 때, 이 보통집합들이 서로 포함관계에 있음을 말한다. Dempster와 Shafer<sup>[5, 6]</sup>의 “상한확률(Plausibility measure)”과 “하한확률(Belief measure)”이 Bayesian Inference 이론에 커다란 영향을 주었으며, 퍼지이론은 바로 “상한확률” 즉, 가장 최악의 경우를 다루는 학문이라고 할 수 있다. 확률이론과 퍼지이론의 관계에 대해서는 Zadeh<sup>[7]</sup> 뿐만 아니라 Dubois와 Prade,<sup>[8]</sup> Yager<sup>[9]</sup>의 석학들에 의해 연구되어 왔다. 확률이론도 분포의 불확실성을 다루지만, 어떤 실험에 대해 확률분포를 이미 아는 경우에만 해당되며, 우리가 임의로 분포를 정해줄 수는 없다. 그러나, 퍼지이론에서는 분포에 대한 가정이 사라지며 평균과 분산등의 모멘트 파라미터 대신에 “가능성분포”라는 확률분포와 유사한 형태를 우리가 임의로 정해줄 수 있는 자유도가 있다. 따라서, 우리는 확률분포를 퍼지집합으로 바꿀 수 있는 반면, 주관적으로 구해진 퍼지집합은 확률분포로 변환은 가능하나 의미를 찾기 어렵다. 일반적으로 확률분포를 퍼지집합의 소속함수로 바꿀 경우, 소속함수는 확률밀도함수에 비해 비교적 넓게 퍼져 있고 포괄적인 모양을 하고 있다. 확률은 사건에 대한 빈도수의 발생을인 반면, 퍼지집합은 발생률뿐만 아니라 모든 가능성을 포함하는 “상한확률”이기 때

## II. 퍼지제어의 기초이론

퍼지논리는 제어에 쉽게 적용할 수 있으며, 이를 퍼지제어라고 한다. 퍼지제어는 1970년대 중반에 영국 Queen Mary College의 Mamdani 교수에 의해 주도되었고, 그의 연구팀은 처음으로 퍼지제어 이론을 실제 시멘트킬린제조공정에 적용하여 성공하였다.<sup>[10]</sup> 이후 일본에서는 Terano, Sugeno 교수 등의 영향을 받아 수많은 퍼지제어 응용제품들이 쏟아져 나왔다.<sup>[11]</sup> 현재 미국에서는 NASA, Rockwell International, General Electric 등, 우주항공 및 방위산업분야에서도 퍼지제어를 적용하고 있다.<sup>[12]</sup> 퍼지제어는 비선형 고차미분방정식을 풀지 않고도 제어시스템의 제어규칙내지는 제어알고리즘을 만들 수 있는 장점이 있다. 다시 말해서, 복잡한 수학적 모델에 의존하지 않고도 전문가의 직감이나 경험상의 법칙, 실제실험 또는 모의실험의 데이터를 통해서 제어에 필요한 규칙들을 유도해 낼 수가 있다. 그런데 기존의 제어기구조에 퍼지전문가시스템을 부착한 형태의 것은 좁은 의미에서의 퍼지제어라고는 하지 않는다. 왜냐하면 최하단부의 제어기가 무엇인가가 중요하기 때문이다. 최근에는 보다 자동화된 기법을 퍼지제어에 적용하는 추세이다. 가변구조제어나 PID제어 등의 기존 제어법을 이용하거나 신경망의 학습방식 또는 유전알고리즘/진화연산 등의 방식을 사용하여 안정된 제어규칙이나 최적의 제어규칙을 만드는 연구가 시도되고 있는데 이는 이른바 “Soft Computing”이라는 큰 범주안에서 각기의 장점을 살린 혼합형의 제어기법이 유리하다고 판단되기 때문이다.

### 1. 퍼지제어를 쓰는 이유는?

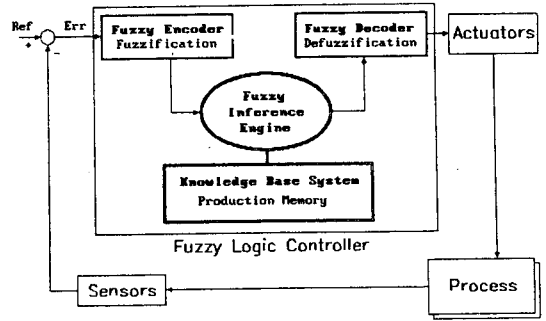
1980년대 중반 이전까지는 제어의 꽃은 적응제어(Adaptive Control)였다. 복잡한 비선형고차미분방정식에 대한 제어기 모델을 사용하였을 때의 위안은 동작영역의 국부화라는 제한적인 가정

이 필요했고, 이것이 이론의 실용화에 걸림돌이 되어 왔다. 그런데, MIT의 Rohrs라는 대학원학생이 선형시스템에서 조차 모델과 실제에 수학적 차이가 있을 경우, 적응제어의 안정성에 심각한 문제점이 발생할 수 있다는 것을 증명하였고, 이것을 기화로 비선형함수를 도입하여 실제시스템을 학습하는 방법과 동작영역에 대한 민감성을 줄이는 방식의 제어가 나타났다. 전자가 신경망제어이고 후자가 바로 퍼지제어이다. 신경망제어는 적응제어알고리즘의 순환적 학습식의 “Sum of Product” 연산에 비선형 임계함수(Threshold Function Unit)를 사용하여 수정한 형태이며, 퍼지제어는 “Maximum of Minimum” 또는 다른 유사한 연산방법으로 불확실한 영역에 대해 매끄럽고 강인한 지능적 추론을 가능케 하는 구조이다. 퍼지제어는 수학적으로 시스템을 표현하기가 복잡하거나 어려운 경우 사용되며, 만약 매우 정확하게 모델링할 수 있는 시스템이 있다면, 이것에 퍼지제어를 사용하는 것은 오히려 성능을 나쁘게 할 가능성이 있다. 그렇다 하더라도 퍼지제어의 장점은 허용오차범위안에서 하드웨어의 도움으로 고속의 결과를 얻을 수 있고 하드웨어가 비교적 쉽게 저가로 구성할 수 있으며 서로 다른 제어대상에 대해 힘안들이고 같은 구조의 제어기를 사용할 수 있다는 점이다. 무엇보다도 중요한 이유는 퍼지시스템이 그 추론과정에 있어서 매우 강인한 특성을 갖는다는 것이다. 그리고, 일반전문가시스템을 사용하여 규칙기반을 만드는 것보다 퍼지시스템을 사용하는 것이 그 자료압축율이 높다는 것도 빼놓을 수 없다.

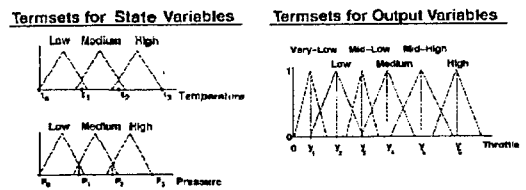
2. 퍼지제어의 기본구조

퍼지제어시스템은 <그림 3>에서 보이는 바와 같이 네 가지 구성요소로 되어있다.<sup>[13]</sup> 퍼지엔코더(Fuzzifier), 퍼지디코더(Defuzzifier), 퍼지추론부(Fuzzy Inference Engine) 및 규칙기반시스템(Knowledge Base System)이다.

퍼지시스템은 변수의 자료유형이 실수일 경우 연속적 퍼지시스템, 정수에 대응되는 경우 이산적 퍼지시스템이라고 부른다. 퍼지제어를 설계할 때는 크게 3단계로 나눌 수 있는데, 첫째 단계에서는



<그림 3> 퍼지제어시스템의 기본구조



<그림 4> 전건부와 후건부의 입출력변수에 대한 퍼지 언어변수

대상시스템의 입출력상태 및 퍼지제어기의 방식을 결정하고 성능지표결정고 퍼지제어규칙의 생성 및 수정보완작업 그리고 강인성과 안정성에 대한 분석이 필요하며, 둘째 단계에서는 코딩작업과 실시간제어를 위한 효율최적화 및 사용메모리 최소화 작업등을 하고, 마지막으로 셋째 단계에서는 퍼지제어시스템을 하드웨어와 소프트웨어로 만드는 과정이 수행된다. <그림 4>에서는 위에서 언급한 첫째 단계에서 전건부를 위한 온도(T)와 압력(P)의 전건부 입력(상태)변수에 대한 퍼지집합의 언어변수들과 후건부 출력변수에 대한 드로틀의 언어변수들의 예를 보인 것이다.

퍼지엔코더 : 퍼지제어의 프로그램을 위한 인터페이스로 측정된 입력정보의 양자화작업이나 퍼지추론을 위한 수치처리문제를 다루는 부분이다. 예를 들어, 이산적 퍼지시스템을 제어기로 선택하였을 경우 모터의 속도나 드로틀밸브의 각도와 같은 자료는 실수이므로 이를 정수영역으로 변환하는 작업이 필요하다.

〈표 2〉 IF-THEN 규칙의 비교 (a) 부울논리의 경우 (b) 퍼지논리의 경우

(a) 부울논리의 규칙 1B 번			(b) 퍼지논리의 규칙 1F 번		
STATE DESCRIPTION		ACTION	STATE DESCRIPTION		ACTION
Temperature(T)	Pressure(P)	Throttle(H)	Temperature(T)	Pressure(P)	Throttle(H)
(T<15) is TRUE	(P<20) is TRUE	OPEN H 80%	T is LOW	P is LOW	H is HIGH

퍼지제어의 규칙기반(Knowledge Base)시스템 :퍼지제어기의 규칙기반은 전건부(입력변수 조건)와 후건부(출력변수 결과)로 나타내며,

RULE NO. k : IF(전건부) THEN(후건부)

로 k번째 규칙을 쓸 수 있다. 퍼지제어에서는 이 부분이 가장 중요하며, 전체 제어시스템의 안정도 및 강인성이 이 퍼지제어규칙을 어떻게 설계하느냐에 달려있다. 가장 쉽게 설계하는 방법은 전문가의 경험과 지식을 동원하여 직관적인 규칙을 만드는 것이다. 〈표 2〉에 부울논리적 규칙과 퍼지논리적 규칙을 예를 들어 비교하였다. 〈표 2〉에서는 온도(T)와 압력(P)에 대해 둘다 낮은 측정값을 가질때 추진에 필요한 연료주입량을 크게 해준다는 퍼지제어규칙을 일반전문가시스템의 부울논리 규칙과 비교한 것이다. 이 경우, 퍼지규칙은 다음과 같다 :

RULE NO. k : IF(T is A) AND (P is B)  
THEN (H is C) (6)

실험자료나 전문가의 조언에 의해 만들어진 전체의 규칙기반의 예가 〈표 3〉에 나타나 있다. 이는 퍼지시스템의 입력인 온도, 압력의 센서자료와 출력인 드로틀 구동장치의 허용동작영역을 상태공간의 표시방법으로 보인 것이다.

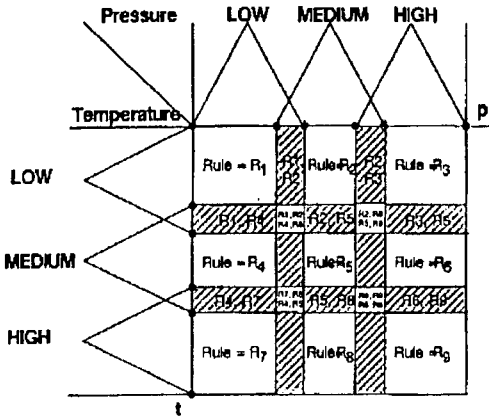
〈그림 5〉에서는 위의 규칙기반을 이용하여 구성된 퍼지상태공간을 보이고 있다. 여기서 빗금친 영역은 삼각형의 퍼지집합을 이용한 규칙들 사이에서 상태공간의 중첩된 부분이다. 퍼지제어시스템의 규칙기반을 구성할 때, 자동화기법을 도입하여 인공지능의 탐색알고리즘이나, 신경망의 학습, 또는 진화연산/유전알고리즘 등을 이용하면 보다 빠르고 효율적으로 퍼지제어규칙을 설계할 수가 있다.

퍼지추론기 : 퍼지추론은 그 방식에 있어서 다양하다. 논리연산에서 Fuzzy(Generalized) Modus Ponens는 Conjunctive AND에 해당하는 Triangular-norm(T-norm)과 Disjunctive OR에 해당하는 Triangular-conorm(T-conorm 또는 S-norm)으로 구성되어 있기 때문에 그러하다. 퍼지추론은 직접법, 간접법, 혼합법의 세 가지로 크게 분류할 수 있다.

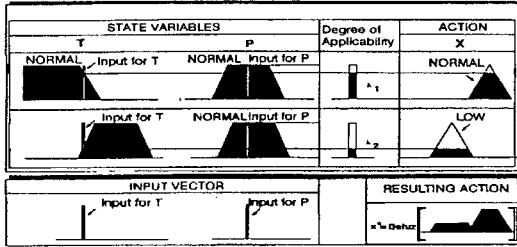
〈표 3〉 규칙기반의 상태공간 표시방법

Rule	Temperature	Pressure	Throttle
R1	LOW	LOW	HIGH
R2	LOW	MEDIUM	MEDIUM
R3	LOW	HIGH	MID-LOW
R4	MEDIUM	LOW	MID-HIGH
R5	MEDIUM	MEDIUM	MOD-LOW
R6	MEDIUM	HIGH	LOW
R7	HIGH	LOW	MID-LOW
R8	HIGH	MEDIUM	LOW
R9	HIGH	HIGH	VERY LOW

Pressure Temp.	LOW	MEDIUM	HIGH
	LOW	R1 Throttle: HIGH	R2 Throttle: MEDIUM
MEDIUM	R4 Throttle: MID-HIGH	R5 Throttle: MID-LOW	R6 Throttle: LOW
HIGH	R7 Throttle: MID-LOW	R8 Throttle: LOW	R9 Throttle: VERY-LOW



〈그림 5〉 온도와 압력에 대해 9개의 규칙으로 구성된 퍼지 규칙기반의 예



〈그림 6〉 Max-Min 퍼지추론방식의 예

(1)직접법 : Max-Min 방식(Mamdani), Max-Dot 방식(Larsen), Sum-Product 방식(Mizumoto)...

(2)간접법 : Baldwin 방식, Tsugamoto 방식...

(3)혼합법 : Sugeno 방식, Simplified 방식...

이중에서 대표적인 것이 Max-Min, Max-Dot 방식이고 실제 적용에 많이 사용되는 것이 단순방식이다. Max-Min 방식은 〈그림 6〉에 사용예가 나와 있다. 여기서 온도와 압력, 드로틀의 i번째 규칙에 대한 퍼지집합의 소속함수를 각각  $\mu_{A_i}(x)$ ,  $\mu_{B_i}(y)$ ,  $\mu_{C_i}(z)$ 라하면 전건부에 대한 적합도는

$$\lambda_i = \mu_{A_i}(x) \wedge \mu_{B_i}(y) \tag{7}$$

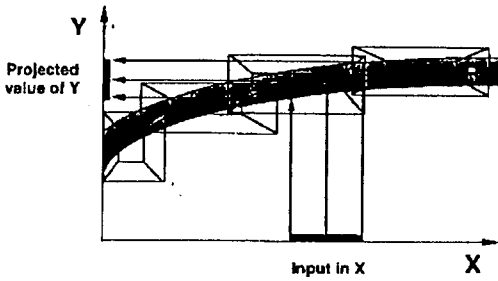
n개의 규칙기반에 대한 Fuzzy Modus Ponens에 의한 퍼지추론의 결과는

$$\mu_c(z) = \bigcup_{i=1}^n \{\lambda_i \wedge \mu_{C_i}(z)\} \tag{8}$$

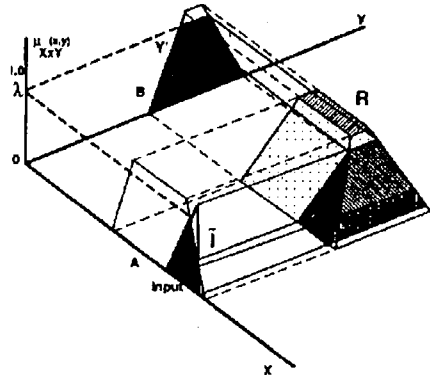
이다. 위 식에서  $\wedge$ 는 T-norm의 일종인 Min 연산,  $\bigcup$ 은 T-conorm의 일종인 Max 연산이다. 〈표 4〉는 각 방식에 대한 퍼지추론의 식을 요약한

〈표 4〉 퍼지추론의 여러가지 방식

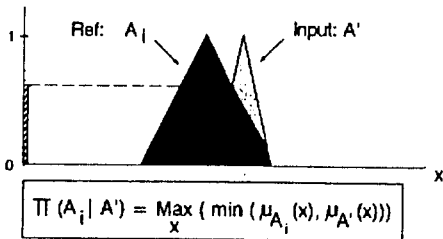
퍼지추론방식	F.M.P.식	규칙의 후건부구조	비 고
Max-Min Mamdani	$\mu_c(z) = \bigcup_{i=1}^n \{\lambda_i \wedge \mu_{C_i}(z)\}$	IF...THEN "z is C"	Defuzzifier 필요
Max-Dot Larsen	$\mu_c(z) = \bigcup_{i=1}^n \{\lambda_i * \mu_{C_i}(z)\}$	IF...THEN "z is C"	Defuzzifier 필요
Sum-Product Mizumoto	$\mu_c(z) = \sum_{i=1}^n \{\lambda_i * \mu_{C_i}(z)\}$	IF...THEN "z is C"	Defuzzifier 필요
Tsugamoto	$z = \frac{\lambda_1 z_1 + \lambda_2 z_2}{\lambda_1 + \lambda_2}$	r1 : IF...THEN N(z <sub>1</sub> ) r2 : IF...THEN P(z <sub>2</sub> )	$z_1 = N^{-1}(\lambda_1)$ $z_2 = P^{-1}(\lambda_2)$ N, P는 단조함수
Sugeno	$z = \frac{\sum_{i=1}^n (\lambda_i * z_i)}{\sum_{i=1}^n \lambda_i}$	rule i : IF...THEN $z_i = a_i x_i + b_i y_i + c_i$	$a_i, b_i, c_i$ 파라미터 추정 필요
Simplified	$z = \frac{\sum_{i=1}^n (\lambda_i * z_i)}{\sum_{i=1}^n \lambda_i}$	rule i : IF...THEN $z_i = c_i$	



〈그림 7〉 단일입출력함수의 Fuzzy Mapping



(a)



(b)

〈그림 8〉 퍼지관계 R과 입력력 X,Y에 대한 추론의 3차원투영

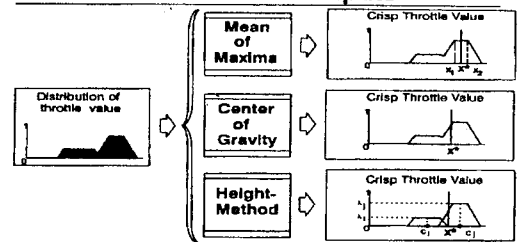
것이다. 이중 간접법과 혼합법은 Defuzzifier의 계산을 줄인 효과가 있다.

〈그림 7〉에 나와 있는 것처럼 단일입출력시스템의 간단한 예를 들어 퍼지추론의 메카니즘을 살펴보면 Fuzzy Modus Ponens는  $Y=f(X)$ 라는 함수에서 X의 신뢰구간에 대해 불확실한 점에 대한 Y의 신뢰구간의 결과를 구하는 문제이며 이것을

“Fuzzy Mapping”이라고 한다. 〈그림 8〉에는 Max-Min 방식의 퍼지추론에 대한 도식적인 해석을 보이고 있고, (a)에서는 투영에 의한 전체적인 추론을 (b)에서는 입력부분에 대한 가능성분포의 퍼지측도를 퍼지관계와의 최소투영의 그래프로 나타낸 것이다.<sup>[14]</sup>

퍼지디코더 : 퍼지추론결과를 프로세스의 제어에 맞도록 수치를 변환하는 작업과 일반 퍼지추론의 경우, 퍼지변수를 Crisp한 실제 값으로 바꾸는 Defuzzification 과정을 포함한다. 이는 최종 퍼지 제어값을 구하는 것으로 여러 가지 종류가 있으나 대표적인 것으로는 Mean of Maxima(MOM), Center of Mass(COM)(연속적 Centroid), Height Method(이산적 Centroid) 등의 방법이 있다. 〈그림 9〉에 이러한 방법들을 비교하였고, 계산식은 다음과 같이 요약할 수 있다 :

**Defuzzification of rule outputs**



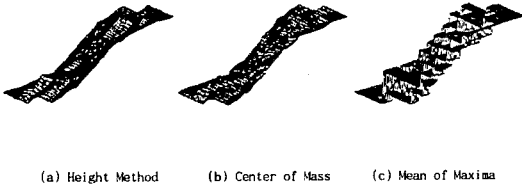
〈그림 9〉 세 가지 Defuzzification 방법의 비교

\* Mean of Maxima :  $z^* = \frac{z_1^* + z_2^*}{2}$  ( $z_i^*$ 는 최대구간의 두 경계점) (9)

\* Center of Mass :  $z^* = \frac{\int z\mu_c(z)dz}{\mu_c(z)dz}$  (10)

\* Height Method :  $z^* = \frac{\lambda_1 C_1 + \lambda_2 C_2}{\lambda_1 + \lambda_2}$  ( $C_i$ 는  $\lambda_i$ 는 중앙값) (11)

위의 세 가지 방법으로 구한 Fuzzy-PD 제어기의 입력출력함수관계인 제어표면을 〈그림 10〉에 비교하여 보이고 있다. (a)는 Height Method, (b)는 Center of Mass, (c)는 Mean of Maxima 방법을 각각 사용한 것이다.



〈그림 10〉 세 가지 Defuzzification 방법에 의한 Fuzzy-PD 제어의 제어표면

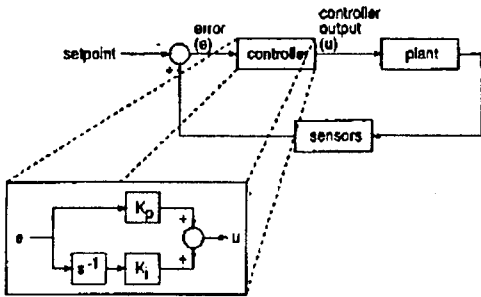
3. 일반 PI제어기와 퍼지 PI제어기의 비교

이제 일반 PI제어기와 퍼지 PI제어기를 비교하여, 퍼지제어기의 다른 장점에 대해 언급하고자 한다. 〈그

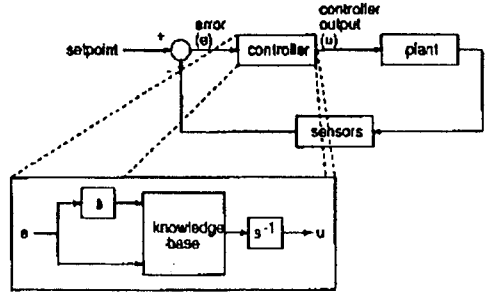
림 11〉에 나타난 것처럼 일반 PI제어시스템의 구조는  $K_p$ ,  $K_i$ 라는 Ziegler-Nichols 파라미터를 Transient-Response 법이나 Stability-Limit 법을 이용해 동조함으로써 제어 특성을 수정하나,<sup>[15]</sup> 퍼지 PI제어시스템은 지식기반시스템을 이용하여 입력단의 오차와 오차의 변화율에 대해 동작영역의 제어표면을 추론해내는 차이점이 있다.

〈그림 12〉 (a)에 나타나 있는 것처럼 일반 PI제어기는 센서자료  $e, \dot{e}$ 를 입력으로 하고, 두 개의 파라미터를 사용하여 제어입력값의 수정능력  $du$ 에 대하여, 3차원 함수공간에서 자유도가 2인 평면을 형성하는데 반해, 퍼지 PI제어기는 〈그림 12〉 (b)에서와 같이 같

**Conventional PI Controllers**

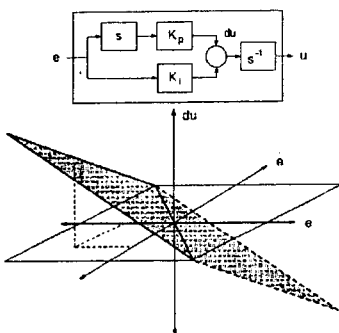


**Fuzzy PI Controllers**



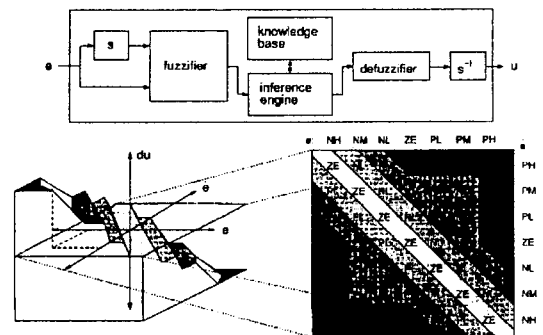
〈그림 11〉 일반 PI제어기와 퍼지 PI제어기의 구조

**Conventional PI Controllers**



(a)

**Fuzzy Logic PI Controllers**



(b)

〈그림 12〉 일반 PI제어기와 퍼지 PI제어기의 제어값 수정능력의 비교



은 센서입력에 대해 사용한 규칙수와  $du$ 의 퍼지 언어변수의 갯수에 비례하여 자유도를 많이 만들어 낼 수가 있다. 바꾸어 말하면, 제어입력에 대한 수정능력이 뛰어나, 점근적 안정도를 높이기 위해 특정 동작영역에서 퍼지규칙을 변화시켜 Transient Response를 좋게 한다든지, Steady-State 상태에서 미세한 오차공간에 대한 퍼지규칙을 많이 만들어서 최소시간에 최소의 오차를 갖도록 설계할 수가 있다는 말이 된다.

4. 퍼지제어는 왜 강인한가?(Sensitivity vs. Robustness)

퍼지제어의 특징은 피제어대상물의 대해 근사적으로 케환된 제어메카니즘을 사용하였기 때문에 파라미터의 변화와 잡음에 대해 별로 민감하지 않도록 되어 있다. 강인한 퍼지추론은 다음의 간단한 예에서 쉽게 알 수 있다. 단일입출력(SISO) 퍼지 연상메모리(FAM)<sup>[16]</sup>가 제어기로 사용될 때,

$$M = \begin{matrix} & 0 & 0.2 & 0.5 & 0.8 & 1 \\ 0.3 & 0.3 & 0.5 & 0.7 & 0.7 \\ 0.7 & 0.7 & 0.5 & 0.3 & 0.3 \\ 1 & 0.8 & 0.5 & 0.2 & 0 \end{matrix}$$

라면, 입력  $A=[1 \ 0.8 \ 0.2 \ 0]$ 에 대한 출력은 Fuzzy Modus Ponens  $B=A \cdot M$ 에서

$$B=[0.3 \ 0.3 \ 0.5 \ 0.8 \ 1]$$

이 얻어진다. 퍼지집합  $B$ 의 이산적 원소에 대한 변수  $y$ 의 대표값이 각각 1, 2, 3, 4, 5라면 Centroid 방식으로

$$y^* = CRISP(B) = \frac{0.3 \times 1 + 0.3 \times 2 + 0.5 \times 3 + 0.8 \times 4 + 1 \times 5}{0.3 + 0.3 + 0.5 + 0.8 + 1} = \frac{6.1}{2.9} \approx 2.103 \quad (12)$$

이 된다. 여기서, 두 가지 파라미터에 대한 강인성에 대해 고찰해 보자.

(i) 퍼지규칙기반 메모리에 대한 강인성 :

설계의 문제점으로  $M$  내부의 파라미터에 가산 오차  $\Delta M$ 이 발생했다고 가정하자. 즉,

$$M' = M + \Delta M = \begin{matrix} & 0 & 0.2 & 0.5 & 0.8 & 1 \\ 0.3 & 0.3 & 0.5 & 0.7 & 0.7 \\ 0.7 & 0.7 & 0.5 & 0.3 & 0.3 \\ 1 & 0.8 & 0.5 & 0.2 & 0 \end{matrix}$$

$$+ \begin{matrix} & 0 & +0.1 & 0 & 0 & 0 \\ & 0 & 0 & 0 & +0.1 & -0.2 \\ + & +0.1 & 0 & 0 & -0.1 & +0.3 \\ & -0.2 & 0 & 0 & 0 & 0 \\ & 0 & 0.3 & 0.5 & 0.8 & 1 \\ = & 0.3 & 0.3 & 0.5 & 0.8 & 0.5 \\ & 0.8 & 0.7 & 0.5 & 0.2 & 0.6 \\ & 0.8 & 0.5 & 0.5 & 0.2 & 0 \end{matrix}$$

일 경우,  $B=A \cdot M'=[0.3 \ 0.3 \ 0.5 \ 0.8 \ 1]$ 이 되어, 여전히  $y^*=CRISP(B) \approx 2.103$ 의 같은 값을 갖게 된다. 이는 일부 파라미터가  $\pm 10\% \sim 30\%$  변화했을 때의 추론결과를 나타낸 경우이며, 퍼지 시스템은 민감도가 적은 파라미터의 큰 변화에 대해서는 강인한 특성을 갖는다.

(ii) 입력자료의 변화에 대한 강인성 :

입력자료에 잡음이 섞여서  $A'=A+\Delta A=[1 \ 0.9 \ 0.3 \ 0]$ 로 변화했다고 가정하자. 이 경우도 퍼지 시스템의 출력은  $B=A' \cdot M=[0.3 \ 0.3 \ 0.5 \ 0.8 \ 1]$ 에서  $y^*=CRISP(B) \approx 2.103$ 의 같은 결과를 얻는다. 따라서, 퍼지시스템의 입력 역시 출력에 대한 민감도가 적은 부분에 일어나는 오차는 결과에 영향을 주지않는 성질이 있다.

5. Stability를 만족하는 퍼지제어시스템의 설계

퍼지제어기를 설계할 때는 항상 안정도(Stability)를 고려하여야 한다. 제어공학에서의 안정성이라는 것은 수학적으로 전체시스템에서 내부의 상태변수가 제한된 입력과 초기조건에 대해 역시 제한된 출력을 갖는 것을 의미하며, 특히 오차에 대하여 점근적인 안정성을 유지하도록 하는 것이 목표이다. 즉,

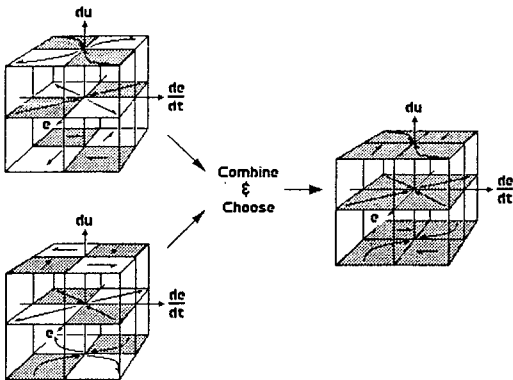
$$e(t) = y_{REF}(t) - y(t) \quad (13)$$

라고 오차를 정의할 경우, 일반적으로

$$\lim_{t \rightarrow \infty} \|e(t)\| = 0 \text{와} \lim_{t \rightarrow \infty} \dot{e}(t) = 0 \quad (14)$$

의 조건을 만족할 때이다. 퍼지제어시스템은 비선

형이므로 Lyapunov의 방법이나 Hyperstability 또는 Bounded-Input Bounded-Output(BIBO) 방법을 적용하는 것이 적당하다. 그런데, 이러한 방법은 안정성을 위한 충분조건이므로 퍼지제어기의 안정도는 얼마든지 다른 방법을 이용할 수가 있다. 예를 들어, 오차를 목표상태로 보내주기 위한 제어 입력조건을 구하여 각 제어입력에 대하여 여러 가지 퍼지분할공간(Fuzzy Partition Space)에서의 궤적을 구한다. 그리고, 이러한 궤적들이 목표상태에 도달하도록 해당 제어입력에 대한 퍼지분할공간을 혼합하여 선택하는 방법이 <그림 13>에 나타나 있다. 이것이 Cell-상태공간을 이용한 Cell-상태전이방법이며 비선형시스템의 Reachability 조건을 이용하여 제어시스템의 안정도가 자연적으로 만족하게 된다.<sup>[17]</sup> 퍼지제어시스템의 안정도는 Lyapunov의 직접법 등과 같이, 제어이론의 여러 가지 안정도를 이용한 방법 등이 있다.



<그림 13>  $(e, \dot{e})$  vs.  $du$  공간에서 두 자료를 혼합하여 만든 퍼지제어분할공간의 안정성

### III. 퍼지제어의 향후 연구과제 및 응용분야

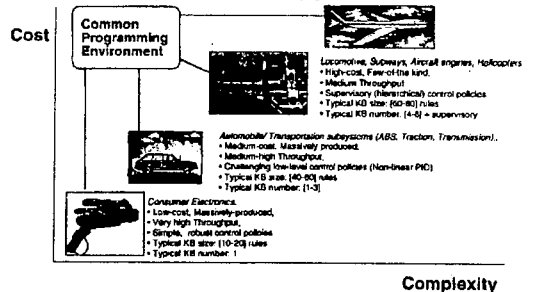
퍼지제어시스템의 장점은 앞에서 언급한 바와 같이, (i) 제어조정능력의 자유도가 높고 (ii) 파라미터변화나 잡음에 대한 민감도가 낮아 견실하며 (iii) 실제적으로 활용하기 쉽고 범용성이 높다는

점 등을 들 수가 있다. 퍼지제어기는 제어기 내부에 동적특성이 없는 경우, 일반적으로 비선형 다변수함수이며 동작영역에 따라 여러 가지 특성을 나타낼 수 있는 비선형제어시스템이다.

그럼에도 불구하고, 퍼지제어규칙들을 유도해 내는 데는 대부분 직관적이고 경험적인 경향이 많기 때문에 불규칙의 범위를 어떻게 정하는가 하는 딜레마에 빠질 우려가 있다. 그러므로 퍼지제어의 장점을 살리면서 퍼지규칙을 학습하는 방법이 필요하고, 현재 신경망을 이용하여 규칙을 학습하는 방법, 유전알고리즘이나 진화연산을 통해 소속함수를 찾아내는 방법, 비선형프로그래밍을 사용하여 규칙을 유도하는 방법, 인공지능의 탐색과정을 거쳐 퍼지집합의 파라미터를 구하는 방법등이 연구중이고 앞으로 개발해야 할 과제이다. <그림 14>는 퍼지시스템응용의 관점에서 퍼지규칙 및 퍼지변수의 복잡성과 퍼지제어시스템 개발비에 연관해서 이미 개발되어 있거나 개발가능한 연구분야와 개발환경에 대한 도표이다.

최근에는 인터넷을 통해서 세계 여러 대학의 연구실과 연구소, 기업에서 퍼지시스템 연구 및 개발의 웹(World Wide Web)서비스를 시작하고 있다. 따라서, 부록에는 인터넷 검색을 통해 찾아낸 퍼지시스템관련 홈페이지 및 FTP 주소와 뉴스그룹을 수록하였다.

### Common Programming Environment for Range of FLC Applications



<그림 14> 퍼지제어응용제품을 위한 프로그램환경

부록 : Fuzzy Logic in Internet, World Wide Web(WWW)

[http : //http.cs.berkeley.edu/projects/Bisc/bisc.welcome.html](http://http.cs.berkeley.edu/projects/Bisc/bisc.welcome.html)  
 Berkeley Initiative in Soft Computing(BISC)  
[http : //ai.iit.nrc.ca/subjects/Fuzzy.html](http://ai.iit.nrc.ca/subjects/Fuzzy.html)  
 Fuzzy logic index in IIT  
[http : //ai.iit.nrc.ca/fuzzy/fuzzy.html](http://ai.iit.nrc.ca/fuzzy/fuzzy.html)  
 Fuzzy CLIPS-Fuzzy logic Expert Systems shell developed at NASA  
[http : //www.cs.tamu.edu/research/CFL/](http://www.cs.tamu.edu/research/CFL/)  
 Center for Fuzzy Logic and Intelligent Systems Research(CFL)-Fuzzy Logic Research Center in Texas A&M University.  
[http : //www.mathworks.com/fuzzytbx.html](http://www.mathworks.com/fuzzytbx.html)  
 MATLAB's Fuzzy Logic Toolbox  
[http : //seraphim.csee.usf.edu/nafips.html](http://seraphim.csee.usf.edu/nafips.html)  
 North American Fuzzy Information Processing Society(NAFIPS)  
[http : //analogy.ie.utoronto.ca/fuzzy.html](http://analogy.ie.utoronto.ca/fuzzy.html)  
 University of Toronto-Intelligent Fuzzy Systems Laboratory  
[http : // www.neuronet.pitt.edu / lcn / research.html](http://www.neuronet.pitt.edu/lcn/research.html)  
 Laboratory for Computational Neuroscience, University of Pittsburgh  
[http : // www.dsi.unimi.it / Users / Labs / LAC / softcomp/index.html](http://www.dsi.unimi.it/Users/Labs/LAC/softcomp/index.html)  
 Soft Computing Group in Milan University  
[http : //life.anu.edu.au/complex-systems/fuzzy.html](http://life.anu.edu.au/complex-systems/fuzzy.html)  
 Fuzzy systems tutorial-complex systems at the Australian National University  
[http : // www.cs.cmu.edu / Web / Groups / AI / html/faqs/ai/fuzzy/part1/faq.html](http://www.cs.cmu.edu/Web/Groups/AI/html/faqs/ai/fuzzy/part1/faq.html)  
 FAQ of fuzzy logic and fuzzy expert systems  
[http : // www.quadralay.com / www / Fuzzy / Fuzzy.html](http://www.quadralay.com/www/Fuzzy/Fuzzy.html)  
 Fuzzy logic archive  
[http : //www.oas.omron.com/](http://www.oas.omron.com/)  
 Omron Advanced Systems-company specializing in factory automation and fuzzy logic

control systems  
[http : //www.digi-net.com/Fuzzy/Home.html](http://www.digi-net.com/Fuzzy/Home.html)  
 Fuzzy Logic : Destination Everywhere-Visit Fuzzy and his friends as they explore space, the net and more!  
[http : // www.yahoo.com / Science / Computer — Science / Artificial-Intelligence / Fuzzy-Logic/](http://www.yahoo.com/Science/Computer—Science/Artificial-Intelligence/Fuzzy-Logic/)  
 Science : Computer Science : Artificial Intelligence : Fuzzy Logic  
[http : // www.cs.cmu.edu / Web / Groups / AI / html/faqs/ai/fuzzy/part1/faq.html](http://www.cs.cmu.edu/Web/Groups/AI/html/faqs/ai/fuzzy/part1/faq.html)  
 FAQ-Fuzzy Logic and Fuzzy Expert Systems  
[http : //viking.cie.cau.ac.kr/homepage.html](http://viking.cie.cau.ac.kr/homepage.html)  
 Laboratory of Fuzzy Systems & Neural Networks at Chung-Ang University  
[ftp : //rtfm.mit.edu/pub/usenet/news.answers/fuzzy-logic](ftp://rtfm.mit.edu/pub/usenet/news.answers/fuzzy-logic)  
 FAQ of comp.ai.fuzzy  
[ftp : //ftp.cs.cmu.edu/user/ai/areas/fuzzy/0.html](ftp://ftp.cs.cmu.edu/user/ai/areas/fuzzy/0.html)  
 Fuzzy logic softwares  
[ftp : //ntia.its.bldrdoc.gov/pub/fuzzy/](ftp://ntia.its.bldrdoc.gov/pub/fuzzy/)  
 Fuzzy logic repository  
[ftp : //ftp.hiof.no/pub/Fuzzy/](ftp://ftp.hiof.no/pub/Fuzzy/)  
 Fuzzy logic collection  
 news : comp.ai.fuzzy  
 Usenet-comp.ai.fuzzy-fuzzy set theory and fuzzy logic.

#### 참 고 문 헌

- [1] L. A. Zadeh, "Fuzzy Sets", Information and Control, vol.8, pp.335~353, 1965.
- [2] L. A. Zadeh, "Outline of a new approach to the analysis of complex systems and decision processes", IEEE Trans. System Man & Cybernetics vol.3, pp.28~44, 1973

- [3] R. J. Larsen and M. L. Marx, *An Introduction to Probability & Its Applications*, Prentice-Hall, 1985.
- [4] G. J. Klir and T. A. Folger, *Fuzzy Sets, Uncertainty, and Information*, Englewood Cliffs, NJ : Prentice-Hall, 1988.
- [5] A. Dempster, "Upper and lower probability induced by a multivalued mapping", *Ann. Math. Statist.*, vol.38, pp.325~339, 1967.
- [6] G. Shafer, *A Mathematical Theory of Evidence*, Princeton, NJ : Princeton University Press, 1976.
- [7] L. A. Zadeh, "Fuzzy logic and its application to approximate reasoning", *Inform. Process*, vol.74, pp.591~594, 1973.
- [8] D. Dubois and H. Prade, "On several representation of an uncertain body of evidence", in *Fuzzy Inform. and Decision Processes*, M. M. Gupta and E. Sanchez, Eds. Amsterdam : North-Holland, 1982, pp.167~181.
- [9] R. R. Yager, "Level sets for membership evaluation of fuzzy subsets", in *Fuzzy Set and Possibility Theory : Recent Development*, R. R. Yager Ed. Oxford : Pergamon Press, 1982, pp.90~97.
- [10] E. H. Mamdani, J. J. Ostergaard and E. Lembessis, "Use of fuzzy logic for implementing rule-based control of industrial processes", in *Advances in Fuzzy Sets, Possibility Theory and Applications*, New York : Plenum, 1983, pp.307~323.
- [11] M. Sugeno, ed. *Industrial Application of Fuzzy Control*, New York : North-Holland, 1985.
- [12] M. Jamshidi, N. Vadiie and T. J. Ross, eds., *Fuzzy Logic and Control*, Englewood Cliffs, NJ : Prentice-Hall, 1993.
- [13] C. C. Lee, "Fuzzy logic control systems : fuzzy logic controller"(part I,II), *IEEE Trans. System Man & Cybernetics*, vol. 20, pp.404~435, Mar/Apr 1990.
- [14] G. Bonnisone, *Approximate Reasoning Systems : Fuzzy Expert Systems and Fuzzy Logic Control*, IEEE Press, 1993.
- [15] F. L. Lewis, *Applied Optimal Control and Estimation*, London UK : Prentice-Hall, 1992.
- [16] B. Kosko, "Fuzzy associative memories", in *Fuzzy Expert Systems*, A. Kandel ed., New York : Addison-Wesley, 1987.
- [17] H. Kang, "Stability and control of fuzzy dynamic systems via cell-state transitions in fuzzy hypercubes", *IEEE Trans. Fuzzy Systems*, vol.1, no.4, Nov 1993.

## 저자 소개



## 姜勳

1959年 11月 19日生

1982年 2月 서울대학교 공과대학 전자공학과 졸업(학사)

1984年 2月 서울대학교 공과대학대학원 전자공학과 졸업(석사)

1989年 9月 미국 조지아공대 대학원 전기공학과 졸업(박사)

(Georgia Institute of Technology)

1989年 9月~1990年 9月 미국 조지아공대 박사후과정 (Research Fellow)

1990年 9月~1992年 2月 미국 조지아공대 연구교수(Research Associate)

1991年 1月~1991年 12月 미국 Automation Concept &amp; Systems 연구원

1992年 3月~1995年 2月 중앙대학교 공과대학 제어계측공학과 조교수

1995年 3月~현재 중앙대학교 공과대학 제어계측공학과 부교수

주관심분야: 퍼지시스템 및 제어, 신경망 패턴인식 및 제어, 로봇트 및 지능제어