

퍼지 추론에 의한 제어방법

변 중 남* · 김 동 화**

(*한국과학기술원 전기 및 전자공학과 교수,

**한국원자력 연구소 선임기술원)

1. 머릿말

주파수 응답을 이용하는 고전적 제어기 설계방법이나 상태 변수공간에서의 시스템 기술을 근거로 하는 현대 제어 방법에서는 플랜트의 수학적 모델링이 매우 중요한 역할을 한다. 그러나 매우 복잡한 프로세스는 수학적 모델을 쉽게 만들 수 없기도 하려니와, 설계사 구한다 하더라도 적절한 제어기를 설계하기가 쉽지 않다. 다시말해서 종래의 Crisp한 제어시스템 설계에서는 설계자가 제어 목표를 설정하고 이를 만족하도록 제어 대상(Plant)에 대한 모델링 및 선형화 작업, 목표치와 제어기구조, 제어정수등을 결정하여 설계한다. 또한 제어기는 처음부터 정해진 평가함수에 의해 목표치와 상태량 X와의 편차를 평가하면서 목표치에 정확히 추종될 수 있도록 제어 명령을 결정한다.

최적 응답, 최소오차를 갖기 위한 제어기 정수의 결정 등은 지금까지 개발된 제어계의 설계 및 시뮬레이션 CAD를 이용해 손쉽게 수행할 수 있으나 그 제어결과가 본래의 제어목적에 어느 정도 만족되는지에 대한 평가는 쉽게 할 수 없는 실정이다. 또한 선형 시스템을 동정했을 경우도 선형작업, 동정오차등이 발생하고 또 동정에 의해 얻어진 시스템일지라도 복잡한 상태변수 및 변수상호간의 관계를 손쉽게 On-line으로 계산할 수 있는 컴퓨터 용량의 한계 및 속도, 가격 등에 의해 실용성에는 상당한 어려움이

뒤따르고 있다. 더우기 시스템 제어시 선형화된 모델일지라도 외란 및 기타여건으로 인하여 순간적으로 비선형으로 변하고 특히 그 구조 자체가 비선형 특성이거나 시스템을 모델링하기 어려운 불명확한 대상, 또는 제어변수가 대단히 많은 선형 시변형과 같은 대상의 제어는 대단히 곤란하다. 물론 상황에 따라 제어정수를 교체한다던가 관측자에 의해 알 수 없는 상태변수를 추정하기도 하나 이의 응용에는 상당한 어려움이 따르게 된다. 또한 열차의 운전제어, 비행기의 이착륙 제어와 같이 운전자의 Know-how 경험에 의한 제어(Expert)가 많은 경우 Man-machine Interface 상호간의 시스템 모델링 및 제어는 사실상 불가능하다고 볼 수 있다.

1970년 초부터 학습제어, 적응제어 또는 지능제어의 이름으로 여러가지 개념이 제안되고 그 응용성이 시험되고 있다. 특히 최근에 컴퓨터의 인공지능분야 발전에 발맞춰 신경회로망과 퍼지이론을 이용한 지능제어의 기법에 대한 관심이 고조되고 있으며 그외에도 여러가지 종류가 제안되고 있으나 여기서는 비교적 Foundation과 Application이 공히 어느 정도 틀이 잡힌 Fuzzy 집합이론을 이용한 제어 방법에 대해 검토하고자 한다.

역사적으로 보면 1965년 퍼지 집합 이론을 제안한 미국의 Zadeh 교수가 1973년 이 이론이 시스템 자동 제어에 응용될 수 있다는 의견을 피력하였으며, 1974년 영국의 Mamdani 교수에 의해 스팀엔진 제

어에 처음으로 응용되어 실용성에의 가능성을 보여주면서 연구에 박차를 가하게 하는 동기가 되었다.

알려진 바와 같이 스티밍엔진의 시스템 동특성은 대단히 비선형 특성일 뿐 아니라 시스템의 모델 또한 어렵고 또 어렵게 모델링하여 얻은 파라미터도 시스템의 출력에 따라 변하게 되어 Crisp한 제어 이론으로는 정확한 제어기 설계, 자동화 응용에 문제점이 있던 것으로 지적되어 왔다. 이에 대해 Fuzzy제어는 선형 특성인 시스템은 물론 비선형 특성, 또는 시스템의 정확한 동특성의 기술이 어려운 시스템에 대해서도 대단히 유용한 것으로 전해지고 있다.

1977년에 Ostergaard는 열교환기와 시멘트 용융로(Kiln)의 Fuzzy 제어에 대한 연구결과 발표를 하였고 이 결과를 발전시켜 1980년에는 덴마크의 스미사는 시멘트 공장의 용융로(Kiln)를 자동화한 제어 시스템을 상용화하였다. 이후 응용부분의 흐름은 일본으로 이어져 퍼지이론 응용의 붐몰이 터치기 시작하였다.

'80년대 들어서는 이의 응용분야가 제어에 국한되지 않고 공학계(엔지니어링 설계, 지능제어, 신호처리, 패턴인식, 이상진단), 사회계(의사결정, 의료, 행동과학, 경제, 사회모델), 자연계(기상, 지도, 생태계, 물리 화학적 현상구명)등 다방면에 걸쳐 유용하게 응용될 수 있다는 것이 입증되어 활발히 연구 중에 있다. 현대제어 이론이 제안된 후 이론적인 연구는 많은 사람들에 의해 수행되어 확립되어 왔으나 그 실용적인 면에 대해서는 시스템 모델링, 시스템의 온라인 계산에 필요한 컴퓨터 용량 및 속도에 대한 제한, 경제성 등으로 인해 지극히 미약한 상태에 있는 것은 부인할 수 없는 사실이다.

이에 반해 Fuzzy 제어는 유용성이 실제 플랜트의 적용에 의해 입증된 후 Linguistic Algorithm의 도출을 제외하고는 실용화가 대단히 간편하여 학문적 이론보다는 오히려 실용화가 앞서있는 실정이다. 이와 같은 여러가지 잇점과 유용성 때문에 유럽, 일본등에서는 이의 관련연구를 위해 연구소의 설립, 국가적 차원에서 연구 프로젝트 설정등 본격적으로 전 분야에 이의 응용 및 관련 연구를 해오고 있다.

일본의 경우 대표적인 예로서 1983년 정수처리장에 퍼지제어이론이 응용되었고 1987년에는 샌다이시의 지하철 제어에 적용되어 에너지 절감과 승차감 향상을 이룬 것이다. 최근엔 가전제품에의 응용이

본격화되어 퍼지 세탁기가 히트하면서 에어컨, 청소기 등 전 가전제품으로 응용이 확대되고 있는 중이다.

이 글월에서는 퍼지 제어 시스템의 핵심개념을 중심으로 구조를 설명하고 퍼지 추론법과 제어기 설계 응용제를 소개한 후 발전된 형태의 퍼지 제어기에 대하여 간략히 기술하고자 한다.

2. 퍼지(Fuzzy) 제어시스템의 구조

Fuzzy Controller의 구조는 그 내부에서 취급되는 정보가 Fuzzy Set이므로 재래식의 제어기와는 매우 다를 것으로 예상된다. 그러나 Fuzzy Controller도 하나의 Controller로서 기본구성이 Software적인 제어측(Control Law)과 Hardware로서의 계산장치(Computing Device)로 나누어지는 기본 골격(그림 1)은 다를 바 없으며, 단지, 제어기 안에 유통되는 정보량의 형태가 다를 뿐이다.

Fuzzy Controller는 (그림 2-(b))에 보인바와 같이

- i) Fuzzy Rule Base
- ii) Fuzzy Inference Engine

을 중심으로 종래의 (Crisp) Numeric Variable의 값을 Fuzzy Set Variable값으로 바꾸거나 그 반대의 작용을 하는

- iii) Fuzzifier
- iv) Defuzzifier

로 구성되어 있다.

이 Fuzzy Controller는 (그림 2(a))에 보인 바와 같이 기능상 Digital Controller와 유사한 구조를 갖고 있음을 알 수 있다. 즉 Digital Variable을 다루는 디지털 제어기에서 A/D Converter나 D/A

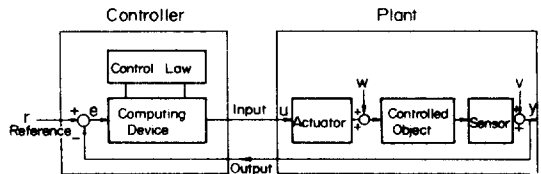
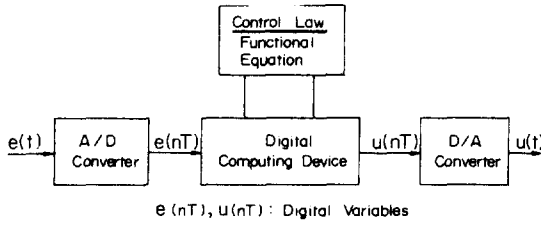
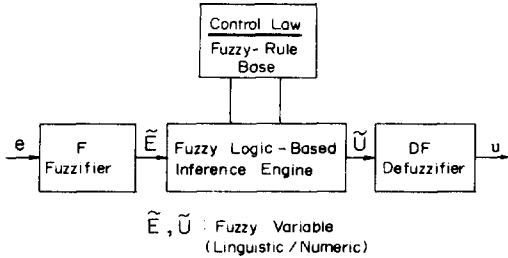


그림 1. Typical control System



(a) Digital Controller



(b) Fuzzy Controller

그림 2. Digital Controller vs Fuzzy Controller

Converter의 역할을 하는 것이 각각 Fuzzifier와 Defuzzifier이며, PID Control Law와 같은 Function Equation 대신에, "If e is \tilde{A} then u is \tilde{B} "와 같은 Fuzzy Rule Base가 쓰이고 있고 Digital Computing Device 대신에 Fuzzy Inference Engine이 계산을 하는 것이다.

이제 구체적으로 이들 각각의 Unit들이 어떤 기능을 하는가 하는 것을 간략히 살펴보기로 하자.

2.1 Fuzzifier

Fuzzifier $F(\cdot)$ 는 종래의 수치적 정보(Crisp Data)를 퍼지집합으로 변환하는 연산자(Operator)이다. 기호적으로 표현하자면

$$\tilde{E} = F(e)$$

로서 e는 Number이고 \tilde{E} 는 Fuzzy Set이다.

제어시스템에 쓰일 수 있는 퍼지화 연산방법으로는 다음과 같이 3가지를 들 수 있다.

(1) Crisp Number를 특수한 Fuzzy Number로 취급하는 법

이것은 주어진 $e = \bar{e}$ 에 대하여 \tilde{E} 를 다음식에

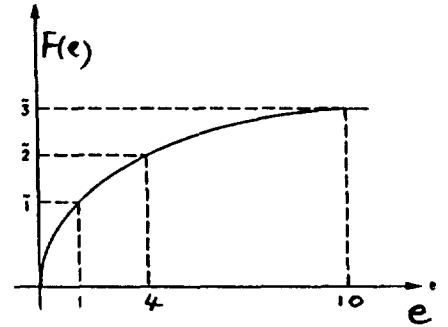


그림 3.

의해 그 소속함수 $m_E(\cdot)$ 구하는 것을 말한다.

$$m_E(e) = \begin{cases} 1 & \text{if } e = \bar{e} \\ 0 & \text{Otherwise} \end{cases}$$

(2) Linear Fuzzy Quantization

이 방법은 먼저 각 정수에 대한 Fuzzy Number를 정한 후 일반적인 수에 대한 Fuzzy Set은 Linear Interpolation에 의하여 구한다. 예컨대 1.7의 퍼지수 1.7은 1과 Fuzzy수 $\tilde{1}$, $\tilde{2}$ 를 이용하여

$$(1 - 0.7)\tilde{1} \oplus 0.7\tilde{2}$$

를 계산한 후 정규화(Normalization)하여 쓴다.

(3) Nonlinear Fuzzy Quantization

제어시스템에서는 오차량을 제거하는 것이 목적이다. 이때 제어입력은 보통 오차에 비례하여 정하게 되나 오차량이 클 경우 제어입력은 포화된 상태값을 취하게 된다. 이를 감안하여 Fuzzy 수로의 변환을 (그림 3)에 보인 바와 같은 비선형 변환을 통해 구하는 방법을 말한다.

2.2 Defuzzifier

Defuzzifier는 퍼지 집합으로 표시된 Fuzzy 양으로부터, 보통의 수치값을 얻는 과정을 나타내는 변환장치이다. Defuzzification하는 방법들을 두가지만 열거하면 다음과 같다. (그림 4 참조)

(1) Mean of Maximum Method(MOM)

Membership Value값이 최대인 변수값들의 평균치를 취하는 방법이다. 이산공간에 정의된 퍼지집합의 경우 $Z_0 = \sum_{j=1}^l \frac{W_j}{l}$ 로서 계산한다. 여기

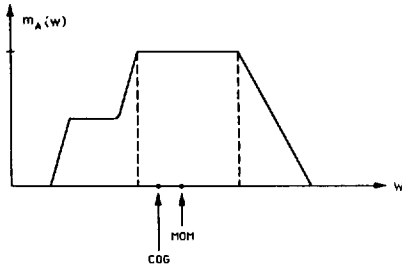


그림 4.

서 W_i 는 소속 함수값이 최대인 점을 나타낸다.

(2) Center of Gravity Method(COG)

가장 많이 쓰이는 Defuzzification 방법으로 서, A라는 퍼지집합에 대하여 Crisp수로서

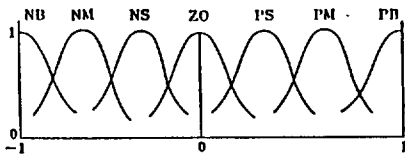
$$Z_0 = \frac{\int w m_A(w) dw}{\int m_A(w) dw}$$

로 계산한다. 이외에도 Possibility Distribution 이 처음으로 최대가 되는점을 취하는 Max Criterion법등이 있으나 설명을 생략한다.

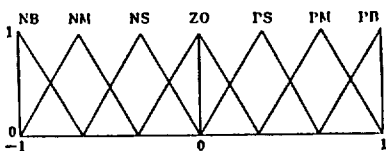
2.3 Fuzzy Rule Base

제어측(Control Law)은 "If-Then-"형태의 Rule로 서 되어 있다. 예를 들어

If e is NB and ce is PB Then n is ZO. else (즉 Error가 Negative Big 이고 Change of



(a) 종 모양의 Fuzzy변수



(b) 3각형 형 Fuzzy변수

그림 5. Fuzzy 변수의 형태

Error가 Positive Big이면 u 는 Zero이다.)

여기서 NB, PB와 ZO는 각각 변수변역에 정의된 Fuzzy Set으로서 전형적으로 삼각형 또는 종형 형태의 소속함수 모양을 갖는다. (그림 5 참조)

동적 시스템을 제어할 때 이러한 퍼지 제어물은 십수개부터 수십개가 될 수 있으며 구할 수 있는 방법도 여러가지가 알려져 있다. 즉,

- (1) Expert의 경험이나 제어공학자의 지식을 이용하여 Rule을 구하는 방법
- (2) 제어 조작자의 동작을 모델링함으로서 구하는 방법
- (3) 제어대상의 Fuzzy Model을 구하고 Fuzzy 시스템 이론을 적용하거나 Simulation을 통해 구하는 방법
- (4) Self-Organizing Controller에서와 같이 학습에 의하여 Rule을 얻는 방법

등이 있다.

2.4 퍼지 추론 장치(Fuzzy Inference Engine)

이 부분은 퍼지룰을 이용하여 퍼지 입력에 대한 퍼지 출력을 추론해 내는 장치이다. 추론하는 형태는 Fuzzy Modus Ponens와 Fuzzy Modus Tollens의 두가지가 있으며 보통 제어에는 전자가, 고장진단에는 후자의 형태를 쓴다.

Fuzzy Modus Ponens는 다음과 같은 논법을 말한다.

Implication(Premise 1) : If e is A , then u is B

Input(Premise 2) : e is A'

Inference(Conclusion) : u is B'

여기서 A, B, A', B' 은 각각 퍼지집합으로 표시되는 양을 나타낸다. 고전 논리학에서는 전제2.에서 $A'=A$ 일 때 결론부가 $B'=B$ 인 경우만 다루며 이를 Modus Ponens(Rule of Detachment)라는 항진 명제라 부르는 것은 잘 알려져 있다. 퍼지 추론 장치에서는 전제2의 변수값이 전제1의 값과 다르더라도(즉 $A \neq A'$) 일반적으로 B' 의 값을 계산해 내는 것이다. 이때 쓰이는 추론 방법에 대하여 다음 장에서 자세히 다루기로 한다.

3. 제어를 위한 퍼지 추론법(Fuzzy Reasoning Methods)

일상 생활에서 흔히 일어나는 다음 예를 생각해 보자.

P1 : 샤워물이 차면 뜨거운 물을 내는 꼭지(Knob)를 연다.

P2 : 샤워물이 차다.

C : 그러므로 뜨거운 물 꼭지를 연다(Open).

이와 같은 언어로 표현되는 명제를 기호로 나타내면

P1 : If x is A , then y is B .

P2 : x is A .

C : y is B .

의 형태로 쓸 수 있으며 이와 같은 추리과정으로, 주어진 조건 명제 P1과 P2로부터 결론 명제 C를 유도하는 과정을 Modus Ponens(Rule of Detachment)라 부른다.

여기서 P2 대신에 다음과 같은 변형된 명제가 주어졌다고 하자.

P2' 샤워물이 미지근하다. (또는 약간 차다)

이 경우 고전 논리로서는 P1, P2'로부터 아무런 결론을 내지 못한다. 그러나 사람은 놀랍게도 뜨거운 물 꼭지를 조금 연다는 결론을 내는 것이다. 이 과정을 공식화하여 써보면 다음과 같다.

(FP1) If x is A then y is B

(FP2) x is A'

(FC) y is B'

(If x is A , then y is B)라는 Implication 전제와 (x is A')이라는 Input 전제로 부터 (y is B')이란 Inference 결론을 끄집어 내는 과정을 퍼지 추론(Fuzzy Reasoning)이라 하며, 일종의 무한치 논리로 확장 발전된 형태로 볼 수 있다. 단지 Fuzzy Logic에는 기존의 논리가 각종 연산자에 대한 닫힘성을 갖는 성질(Completeness)이 없기 때문에 이를 근사 추론(Approximate Reasoning)이라 부르기도 한다.

현재까지 제어시스템과 관련하여 이용되고 있는 퍼지 추론법을 직접법, 간접법과 혼합 규칙법으로 나누어 설명하기로 한다.

3.1 직접법(Direct Fuzzy Reasoning)

이 방법에서는 Inference 결론부를 퍼지 수학적으로 직접 계산하는 방법이다. 즉 주어진 implication

과 Input에 대하여 결론부 Inference를 합성 연산자(Composition Operation)에 의하여 얻는다.

Implication : $A \rightarrow B$

Input : A'

Inference $B' = A' \circ (A \rightarrow B)$

Fuzzy Set의 합성 연산법과 Implication의 소속 함수 계산법이 여러가지가 있기 때문에 구체적으로 B의 값을 구하는 방법은 여러가지가 있을 수 있다.

예컨대 Implication으로서 $A \rightarrow B$ 를 $A \times B$, $\bar{A} \cup B$, $(A \times B) \cup (\bar{A} \times Y)$ 등으로 대체하는 방법이 있고 Composition $A \circ B$ 를 Max-Min 또는 Max-Product로 계산한다. 여기서는 퍼지 제어에 많이 쓰이는 두가지 방법을 예를 들어 설명하기로 한다.

3.1.1 Mamdani의 최소연산측(Minimum Operation Rule)

영국 Imperial College의 Mamdani 교수는 Inference에 있는 관련 연산자를 매우 단순화하여 처리하는 방법을 다음과 같이 제안하였다.

$$B' = A' \circ (A \rightarrow B)$$

$$= A' \times (A \times B)$$

즉

$$\max_x \{m_{A'} \wedge [m_A(x) \wedge m_B(y)]\}$$

여기서 \wedge 은 Minimum Operator이다.

윗 식에서 우리는 $m_B(y)$ 가 보다 쉽게 계산됨을 알 수 있다. 즉,

$$m_B(y) = [(m_A(x) \wedge m_A(x))] \wedge m_B(y)$$

따라서

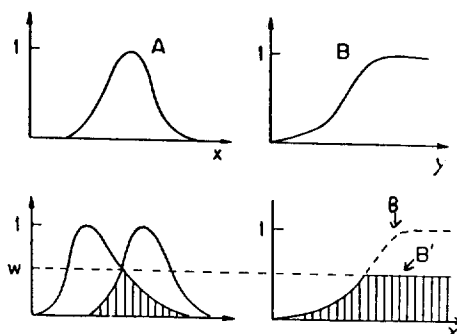


그림 6. Graphic Computation of Mamdani's Method

$$\alpha = \max_x \{m_A(x) \wedge m_A(x)\}$$

라 놓으면

$$m_B(y) = \alpha \wedge m_B(y)$$

로 간단히 계산된다. 이때 α 를 A와 A'의 적합도 (Degree of Compatibility)라 부른다. (그림 6)에 이 연산방법에 대한 도식적 설명이 주어져 있다.

3.1.2 Larsen의 승법연산속(Product Operation Rule)

Larsen은 Implication 합성 연산자를 Product 연산으로 처리하는 방법을 제안하였다. 즉 A와 A'의 적합도 α 에 대하여 B'의 소속 함수 계산은

$$m_B(y) = \alpha \cdot m_B(y)$$

로 하는 것이다.

3.1.3 예 제

예를 들어 퍼지 제어물이 2개이며 입력 변수가 둘인 경우를 생각해보자.

R_1 : If x is A_1 and y is B_1 , then $Z = C_1$

R_2 : If x is A_2 and y is B_2 , then $Z = C_2$

만약 $x = x_0$ 이고 $y = y_0$ 일 때(그림 7)은 Mamdani의 방법에 의한 추론결과이다.

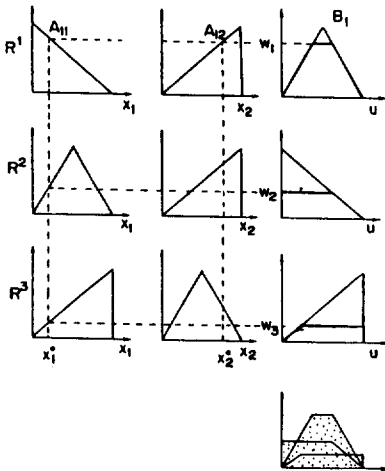


그림 7

3.2 간접법

이 간접법에서는 x is A라는 명제와 x is A'라는 명제를 비교해서 A가 A'이 된 것은 x is A란 명제의 실제적인 참의 정도가 절대참은 아니기 때문이라는 전제하에 진리값을 나타내는 Fuzzy Set τ_A 를 도입하여

$$"x \text{ is } A" \text{ is } \tau_A \iff x \text{ is } A'$$

란 등식관계로 부터 간접적으로 유추하여 " y is B"라는 결론이 " y is B"이 되는 관계를 얻어내는 방법이다. 구체적으로 다음의 3단계를 거쳐 B'을 계산한다.

단계 1. : A와 A'이 주어진 상태에서 τ_A 를 계산한다.

$$m_{\tau_A}(x) = m_A(m_A^{-1}(x))$$

단계 2. : τ_A 가 주어진 상태에서 $\tau_{A-B} = 1 \wedge (1 - \tau_A + \tau_B)$ 로 부터 τ_B 를 계산한다.

이때 τ_{A-B} 는 보통 True로 놓는다.

단계 3. : " y is B" is $\tau_B \iff y$ is B'이란 등식관계로 부터 B'을 구한다.

$$m_{B'}(y) = m_{\tau_B}(m_B(y))$$

3.2.1 Tsukamoto's Method

이 간접 추론법을 단조 소속 함수를 갖는 Fuzzy 변수에 대하여 알기 쉽게 제어에 적용하는 방법을 Tsukamoto가 제안하였다. 3-1.3의 예제에 대하여 생각하면 Rule 1과 Rule 2에 적용하여 구한 적합도 α_1 과 α_2 에 대하여

$$\alpha_1 = C_1(y_1)$$

$$\alpha_2 = C_2(y_2)$$

의 관계식으로 부터 y_1 과 y_2 를 구한 후

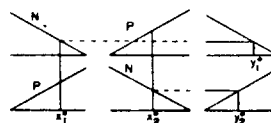
$$Z_0 = \frac{\alpha_1 y_1 + \alpha_2 y_2}{\alpha_1 + \alpha_2}$$

에 의하여 Z_0 를 계산한다.

3.2.2 예 제

R^1 : IF x_1 IS N_{11} AND x_2 IS P_{12} THEN y IS N

R^2 : IF x_1 IS P_{21} AND x_2 IS N_{22} THEN y IS P



$$\begin{aligned} \omega_1 &= N(x_1^0) \wedge P(x_2^0) \\ \omega_2 &= P(x_1^0) \wedge N(x_2^0) \\ y_1 &= N^{-1}(\omega_1) \\ y_2 &= P^{-1}(\omega_2) \\ y^0 &= \frac{\omega_1 y_1 + \omega_2 y_2}{\omega_1 + \omega_2} \end{aligned}$$

$$\begin{aligned} dE &= E_n - E_{n-1} \\ &= (R - Y_n) - (R - Y_{n-1}) \\ &= Y_{n-1} - Y_n \\ dU &= U_n - U_{n-1} \end{aligned}$$

3.3 혼합형 조건 명제에 의한 추론법

이 방법은 일본의 Takagi 및 Sugeno에 의하여 제안된 것으로 제어 법칙이 다음과 같이 조건부는 Fuzzy 변수에 의하여 기술되고, 후건부는 Crisp 관계에 의하여 기술되는 Rule Base로 표시되는 경우를 다룬다.

$$R_i : \text{If } x \text{ is } A_i \text{ and } y \text{ is } B_i, \text{ then } y = f_i(x, y)$$

예를 들어 2개의 Rule이 있는 경우 $x = x_0, y = y_0$ 이 입력될때의 Z_0 는

$$Z_0 = \frac{\alpha_1 f_1(x_0, y_0) + \alpha_2 f_2(x_0, y_0)}{\alpha_1 + \alpha_2}$$

에 의하여 계산한다.

4. Fuzzy 제어기 설계에 관하여

4.1 설계 예

Fuzzy 제어를 실제에 응용할 때 문제가 되는 것은 Fuzzy 제어규칙을 어떻게 만드는가 하는 것이다. 이 절에서는 구체적인 예를 보이기로 한다. (그림 8)과 같은 응답이 예상되는 단일 입출력 제어계를 생각하자. 제어규칙은 형식의 직접법을 쓰기로 한다. "If" 파트의 전건부 변수로서 시간 nT 에 있어서 출력은 Y_n , 편차를 E_n , 조작량 U_n , 설정치 R 이라 하고 증분 dE 와 dU 를

라 하자. dE 는 출력 Y 의 구배의 부호를 반대로 한 것으로 출력이 상승하였을때는 $dE < 0$, 하강하였을때는 $dE > 0$ 이다.

Fuzzy 제어기의 입출력 관계는 $E, dE \rightarrow U$ 로 표시하기로 하며, 이것은 U 의 시간미분 즉 속도에 상당하는 dU 를 출력으로 하는 제어 Algorithm으로 속도형이라 부른다. 제어기 구조로서 U 의 값을 후건부 변수로 하는 위치형을 채용할 수가 있으나, E 의 적분값을 후건부 변수로 하여야 하므로 계산이 대단히 복잡하고 속도형의 방법이 제어규칙의 수가 적어도 되는 잇점이 있다.

(그림 8)에서 알 수 있는 바와 같이 이 응답은 4가지 형태 I, II, III, IV의 반복으로 구성되어 있고 1 Cycle에 비해 2 Cycle에서는 Plant 출력의 크기가 감소한다.

우선 각 Cycle의 특징적인 점을 선정, 요약하여 Fuzzy 제어규칙의 전건부가 특징점을 기술하도록 한다. 예를 들면 1 Cycle Phase I의 a1 부근에서는 편차(설정치-출력)는 (+)로 크고, Plant출력은 거의 입상하지 않았으므로 dE 는 영에 가깝다.

이것을 $E = PB$ and $dE = ZO$ 와 Fuzzy변수의 값을 이용하여 기술한다. 이 부근에서는 당연히 조작량은 가장 크게 $U = PB$ 로 할 수 있다. 같은 방법으로 Phase II의 점 b1, III의 c1, IV의 d1등의 부근에 관해서도 제어규칙을 만들면

- a1 : if $E = PB$ and $dE = ZO$ then $dU = PB$
- b1 : if $E = ZO$ and $dE = NB$ then $dU = NB$
- C1 : if $E = NB$ and $dE = ZO$ then $dU = NB$
- d1 : if $E = ZO$ and $dE = PB$ then $dU = PB$

이 된다.

2 Cycle의 a2, b2 등의 부근에서는 a1, b1에 비해 E 또는 dE 의 절대치가 적게되므로 그에 따른 값도 적게하면 좋다. 즉,

- a2 : if $E = PM$ and $dE = ZO$ then $dU = PM$
- b2 : if $E = ZO$ and $dE = NM$ then $dU = NM$

이 된다. 표1은 이와 같이 해서 만든 13개의 규칙을 나타낸 것으로 종축은 E 의 값, 횡축은 dE 의 값이고 표중에서는 dU 값을 나타내고 표1(b)는 조건부가 정

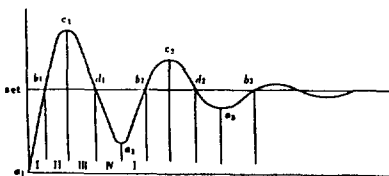


그림 8. 플랜트 출력의 시간응답

표 1. 제어규칙표

		dE						
		NB	NM	NS	ZO	PS	PM	PB
E	NB						NB	
	NM						NM	
	NS						NS	
	ZO	NB	NM	NS	ZO	PS	PM	PB
	PS						PS	
	PM						PM	
	PB						PB	

(a)

		dE						
		NB	NM	NS	ZO	PS	PM	PB
E	NB						+6	
	NM						+4	
	NS						+2	
	ZO	+6	+4	+2	0	-2	-4	-6
	PS						-2	
	PM						-4	
	PB						-6	

(b)

수인 경우의 제어규칙을 나타낸다.

표1에서 알 수 있는 바와 같이 규칙의 최대수 $7 \times 7 = 49$ 에 대해서 실제로 있는 것은 13개로 거의 다른 개소는 공백이다. 이와같이 적어도 되는 것은 1개의 규칙이 커버하는 영역이 넓고 현실적으로 규칙을 얻을 수 없는 영역(예를들면 $E=PB$ and $dE=NB$)에서는 규칙이 필요하지 않기 때문이다.

4.2 Fuzzy제어의 정수장에 응용

●공정의 개요

취수지에 도착한 물은 철, 망간, 기타 오물을 제거하기 위해 염소를 주입하여 소독 및 침전효과의 증대를 꾀하는데 이때 염소의 주입량은 물에 포함된 여러가지의 이 물질량에 따라 변동한다. 또한 물중의 염소는 침전지나 여과지에서도 소비되고 기 후, 주·야간, 물의 처리량에 따라 잔류염소 농도도

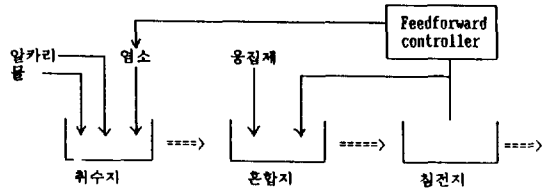


그림 9. 정수장 프로세스 약도

변한다. 즉 정수장의 염소주입 제어는 외란요건이 대단히 많고 반응효과 시간이 길어 피드 백 제어가 곤란하여 6-8시간 전에 미리 외란요건을 고려하여 예측 제어를 하기 위한 Feedforward제어가 필요하다.

●종래의 염소주입 제어

침전지 입구의 잔류염소 농도를 측정(일조량, 햇빛의 강도등을 고려), 기후, 취수량등에 따라 판단 기준을 다르게 만들어 이를 보고 염소량을 산정한다.

이러한 과정은 계산기에 의해 데이터화 되어 있으나 1일 단위로 계산되어 있기 때문에 돌발적인 기후 변동이나 취수량 변동에는 예측이 대단히 제어가 어렵다.

●Fuzzy제어의 구현

숙련된 운전원은 계절, 기후, 물의 취수량에 따라 염소 농도를 산정하므로 이를 요약하며 다음과 같이

표 2. 제어규칙

구분	적 요					
	취수량	계절	시간	기후	기준목표치	보정치
제어규칙 1	적	다	심야		꽤적게	
	"	"	아침		조금적게	
	"	"	오전		조금많게	
	"	"	오후		적게	
	"	"	밤		꽤적게	
제어규칙 2	"	봄	오전	맑음		조금많게
	"	여름	"	"		꽤많게
	"	가을	"	"		조금많게
	"	봄	"	"		조금적게
	"	여름	"	"		그대로

제어규칙을 만든다. 표2는 제어규칙의 한 예이다.

- i) 계절 : 일조량 판단기준으로 춘, 하, 추, 동 4 단계
- ii) 기후 : 염소소비의 일조량 영향에 대한 판단기준으로 맑음, 구름, 비 3단계
- iii) 원수 취수량 : 물의 체류시간 변동에 따라 염소 소비 판단기준으로 적다, 보통, 조금 많다, 많은 4단계
- iv) 시각 : 기후와 함께 일조량에 따른 염소소비 판단기준으로서 밤, 아침, 오전, 오후, 심야 5 단계
- v) 기준목표치 : 꽤 낮다, 낮다, 조금 낮다, 보통, 조금 높다, 높다, 꽤 높다는 7단계
- vi) 보정량 : 상당히 적게, 적게, 조금 적게, 그대로, 조금 많게, 많게, 꽤 많게의 7단계

이들의 제어규칙은 If-Then의 형식으로 나타내어 입력변수명(취수량을 QIN으로 표시)과 그의 Fuzzy 변수명(적다를 SA로 표시)을 대립시켜 조건부 명제로 하고 출력변수명(보정량을 DCL로 표시)과 그 Fuzzy 변수명(조금 많다는 ML로 표시)를 대립시켜 후전부 명제로 하여 입력시킨다. 취수량은 그날의 취수량을 기후는 운전원에 입력치를, 계절·시각은 계산기가 자동적으로 계산한 값을 입력값으로 한다. Fuzzy제어기는 기준목표치와 그의 보정치들 구해 Fuzzy추론을 각 1회 수정하여 얻어진 결과를 가해서 침전지 입구 잔류염소 목표치와 염소 주입용 제어기의 설정치를 낸다.

●Fuzzy제어 효과

이와 같이 해서 얻어진 Fuzzy제어의 효과는 기후의 입력만으로 목표값이 얻어지므로 운전원의 부담이 경감되고 운전원 훈련에 필요한 시간과 1일의 작업량이 대폭 경감된다.

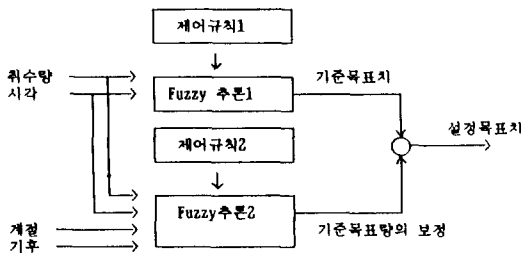


그림 10. 제어기 기본구조

4.3 Fuzzy제어 H/W 및 S/W

Mamdani가 Fuzzy 집합론을 실제의 플랜트에 응용할때 이용한 컴퓨터는 PDP-8이었고 이어 F.L. Smith는 16비트 제어용 컴퓨터와 Fuzzy 추론용 S/W를 이용해 시멘트 킬른 자동 제어의 문제점을 대폭적으로 개선하였다. 1980년 전후에는 일본의 기업들(日立製作所, 富士電機)이 여러방면의 제어 시스템에 Fuzzy제어를 응용하였으나 이때까지의 응용기법은 기존의 제어용 컴퓨터와 Fuzzy 추론용 S/W를 이용한 것이었다. 세계 최초로 Fuzzy 추론용 전용칩이 개발된 것은 1985년으로 AT&T Bell연구소의 Togai씨 등에 의해서이다. 이들은 Max과 Min의 기본적인 Fuzzy이론 연산을 이용해 Fuzzy 추론용 VLSI Chip을 개발하였는데 Membership 함수의 값 [0, 1]을 4bit로 이산 표시한 CMOS Digital Chip으로서 Max, Min 합성의 Fuzzy 추론을 8만 FLIPS (Fuzzy Logical Inference per Second)로 할 수 있는 능력을 갖고 있고 1칩당 16롤을 실행할 수 있는 것으로 알려지고 있다.

이어 일본의 Yamakawa교수가 기관상에 개별부품으로 조립한 것을 1개 칩으로 한 Fuzzy 추론용 Chip, 히로다가오루 씨와 Micon사가 1988년 8월에 개발한 Fuzzy 추론을 메모리화한 방식을 발표하여 1989년 2월에 Fuzzy 개발 기본용으로 시판하고 있는 것들이 있다.

Fuzzy 추론용 S/W에 대해서는 이미 기술한 바와 같이 80년대 초반까지 Fuzzy 제어를 응용한 것은 기존의 제어용 컴퓨터에 Fuzzy 추론 S/W를 응용한 것이었다. Fuzzy제어용 S/W는 현장 지식정보에 해당하는 전전부와 관측정보의 조합에 의한 후전부 정보에서 Fuzzy추론 출력을 얻고 그 출력에서 확정치를 결정하며 비 Fuzzy화를 수행하므로써 최종 추론 결과를 얻게된다. 이러한 과정을 프로그램으로 기술하는 경우 Fuzzy 연산, Fuzzy 추론 합성규칙, 비 Fuzzy화 연산, Membership 함수의 표현방식(정수 또는 소수), Fuzzy 제어알고리즘등을 어떻게 하느냐에 따라 여러 프로그램이 나올 수 있다. 이러한 관점에 따라 개발된 Fuzzy 제어용 S/W에는 영국의 J. F. Baldwin Bridge 대학 교수가 개발한 FRIC(1987년부터 시판중인데 Fuzzy 관계 데이터 베이스를 구축하는 Tool로서 퍼스컴에서 중형 컴퓨터까지 이용

표 3. RPX-Fuzzy사양

구분	적 요
· 적용O.S	MS-DOS(Package ORPX)
· 사용디스크	3.5/5" FDD, 520KB
· 적용H/W	IBM FA
· 추론법	전향 Fuzzy 추론
· 제어규칙	If-Then 60조 건/16결론, 최대 500 규칙
· Fuzzy label	입력변수용 5종, 출력변수용
· 출력알고리즘	속도형
· 모니터링기능	ORPX표준기능
· 튜닝기능	온라인 오프라인가능
· 가격	약250만엔

표 4. 明電Fuzzy제어 시스템 사양

구분	적 요
· 제어기	i8086+u port (2MB) 4MB HDD, 1MB FDD
· Man-Machine	14" CRT, Keyboard, Mouse
· Process I/O	PIO port 내장, 별도의시퀀서, 80입력/48출력
· 제어주기	0.1-999.9초 가변
· 추론법	Max-Min 합성, 중심법
· 제어규칙	If-Then, 4조건/2결론, 최대 250 규칙
· Fuzzy-label	15종
· 멤버십 함수	이산형
· 출력알고리즘	속도, 위치형
· 모니터링기능	추론상태의 도형표시
· 튜닝기능	규칙, 멤버십 함수의 온-라인 변경

가능하다.) Blue Circle사와 LINK man, 일본 富士電機의 FRUITAX OC engineering, 明電舎, 미스비시 전기 등 기업적 차원에서 개발하여 상품화 하고 있는데 어느것이고 16비트 기종으로 사용되고 있고 제어주기는 0.1초정도 이상인 것으로 알려지고 있다.

이외에 미국 Decision Products사의 의사결정 지원 시스템(Max-Min 합성중심) 알고리즘에 기초를 두고 있다. W. Silver교수의 Kemp carry 심장 연

구조에서 이의 응용에 관점을 두고 개발한 FLOPS 가 있다.

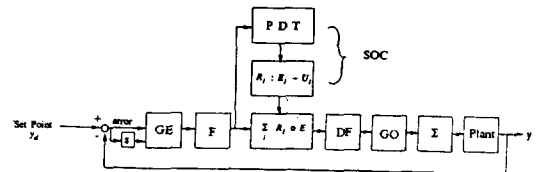
후자는 IBM PC상에서 심전도의 자동분석을 실시 할 목적으로 개발된 것으로 그 결과가 양호하여 시 판중이다. 대표적인 이의 사양을 요약하면 표3과 같다.

그외에 퍼지 제어기로서 개발되어 있는 제품으로는 FRUITAX(Fuzzy Rule Information Processing Tool for Advanced Control System), Fuzzy 제어 시스템, FZ-3,000, 5000, LINK-man등이 있다(표4 참조).

5. Advanced Fuzzy Controllers

5.1 자기 구성 제어기(Self-Organizing Controller)

(그림 11)에 보인 자기구성 제어기(약칭 SOC)는 숙련자의 경험이나 사전지식으로 Rule을 생성하기 어려울때 실제 수행중에 얻어진 정보들을 토대로 제어기 자체가 새로운 룰을 생성 또는 개선시킴으로서 원하는 제어성능을 얻을 수 있는 개념으로서 Procyk과 Mamdani에 의해 제시되었다. 이는 일종의 적응 퍼지 제어기라 볼 수 있다. SOC의 특징은 제어계의 성능을 평가하여 그 평가에 따라 룰을 변경 또는 생성시킬 수 있는 Performance Decision Table(PDT)(그림 12 참조)를 기준으로 하여 룰을



- GE : Input Gain
- F : Fuzzifier (Fuzzy Quantizer)
- PDT : Performance Decision Table
- DF : Defuzzifier
- GO : Output Gain
- Σ : Summation
- (Velocity form - Position form)

그림 11. 자기 구성 제어기(soc)

		Change in error						
		NB	NM	NS	ZE	PS	PM	PB
Error	NB	-60	-60	-40	-40	-20	-20	0
	NM	-60	-40	-40	-20	-20	0	20
	NS	-40	-40	-20	-20	0	20	20
	ZE	-40	-20	-20	0	20	20	40
	PS	-20	-20	0	20	20	40	40
	PM	-20	0	20	20	40	40	60
	PB	0	20	20	40	40	60	60

그림 12. Performance Decision Table

변경시킨다는 것이다. 이 Table에는 오차와 오차변화가 대각선상에 놓여 있는 Zero에 해당되는 상태로 변화하도록 하는 추가 입력량들을 표현하고 있다. 구체적으로 $e(nT)$ 와 $c(nT)$ 를 이용하여 PDT로부터 입력량에 보상해 주어야 할 보상량 P 를 찾는다.

$$P = PDT\{e(nT), c(nT)\}$$

윗 식에서 $e(nT)$ 와 $c(nT)$ 는 제어대상의 지연시간 (mT)만큼 과거의 제어입력 $U(nT - mT)$ 의 영향으로 볼 수 있다. 그래서 보상된 량을 $V(nT - mT)$ 라 놓고

$$V(nT - mT) = U(nT - mT) + P$$

라는 등식을 만든다. If ($E(nT - mT)$, $C(nT - mT)$), Then $U(nT - mT)$ 라는 룰에 해당하는 관계 행렬

$$R'(nT) = E(nT - mT) \times C(nT - mT) \times U(nT - mT)$$

을 IF($E(nT - mT)$, $C(nT - mT)$) Then $V(nT - mT)$ 라는 룰 즉,

$$R''(nT) = E(nT - mT) \times C(nT - mT) \times V(nT - mT)$$

로 고쳐 놓는다.

(그림 11)의 자기 구성 제어기에서는 또한 Input/Output Gain GE 및 GO를 적절히 선택하여 성능향상을 도모할 수 있는 자유도가 있다.

5.2 퍼지 뉴로 제어기(Fuzzy-Neuro Controller)

지난 1990년 7월에 일본에서 Fuzzy Logic and Neural Network에 관한 국제학술회의가 있었으며, 이 회의에서 한 중요한 주제가 Fuzzy-neuro System (혹은 Neuro Fuzzy System)에 관한 연구의 방향에 관한 것이었다.

지능시스템의 두가지 필수요건으로서

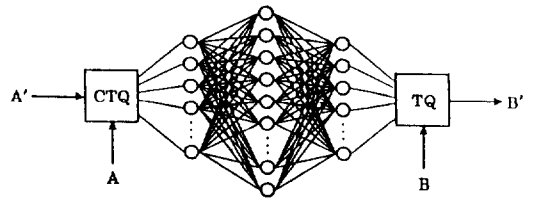


그림 13. Structure of fuzzy-inference neural network

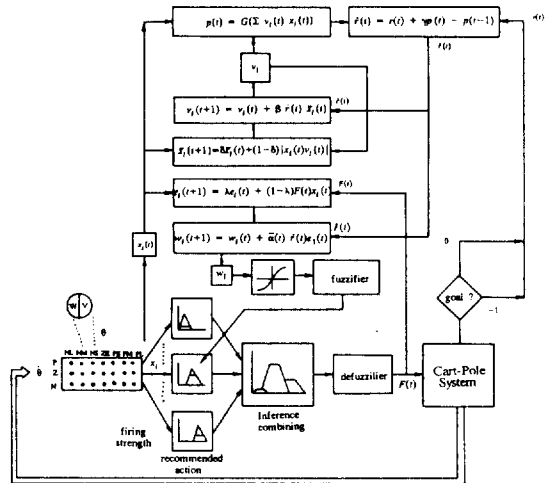


그림 14. Signal flow of the intelligent control system

- i) 불확실성의 취급기능
- ii) 학습기능

을 꼽고 있다. 잘 알려진 바와 같이 인공 신경 회로망은 학습기능을 갖추고, Fuzzy-Logic관련 시스템은 애매하고 불확실한 정보를 잘 처리할 수 있으므로 Fuzzy Logic과 Neural Net을 결합하여 두 요건을 모두 갖춘 기능 제어기를 구상하는 것은 당연히 기대할 수 있는 것이다. 어떤 방식으로 결합하는가 하는 것이 문제이다.

이와 관련하여 현재까지 발표된 연구논문중 Fuzzy Inference를 구하는 과정에 신경회로망의 학습 조정기능을 도입하여, 훈련(Training)에 의하여 보다 성능이 향상된 Fuzzy Inference를 얻고자 하는 노력이 두드러져 보인다.

(그림 13)에 간접법에 의한 추론 과정에 응용된 구조를 보였으며 (그림 14)에 구체적으로 제어시

시스템에 응용된 예를 보였다.

6. 결 론

지금까지 퍼지 논리를 이용한 제어시스템에 관하여 핵심 개념을 중심으로 기술하였다. 요약컨데 이 퍼지제어기의 특징은

- (1) Parallel (Distributed) Control
- (2) Logic Control
- (3) Linguistic Control

등이며, 퍼지 제어가 효과적일 수 있는 제어대상 (Plant)로서는 수학적 모델을 적용하기 힘든 시스템으로서 경험적으로 또는 수동적인 방법으로 제어가 잘되고 있는 대상을 들 수 있다. 그 뿐 아니라 간단한 제어가 필요한 경우로서 보다 효과적인 제어측 Software를 쓰거나 센서 또는 필터없이 사용가능하고, Inverted Pendulum의 자세 제어처럼 정확성보다는 속도 응답 제어가 요구되는 경우 등에 효과적으로 쓸 수 있는 것으로 알려지고 있다. Fuzzy 제어는 지식 베이스의 규모에서 인공지능형 Expert

System보다 Compact하고 선형·비선형 플랜트에 공히 이용될 수 있으며, 설계자는 오퍼레이터와의 접촉을 통해 룰을 구축하므로 사용자가 시스템을 이해하기 쉬운 잇점등이 있기도 하다.

그러나 가장 큰 문제는 구축해 놓은 시스템의 안정성(Stability)를 이론적으로 사전에 검증하기가 어렵고, 같은 제어대상이라 할지라도 추론방법, 소속함수의 형태선택, 룰수 등에 따라 제어성능이 바뀔 수 있으나, 무엇이 어떤 영향을 주는지 규명되지 않은 점 등 여러가지 연구되어야 할 내용이 많이 있다.

참 고 문 헌

- [1] M. Sugeno, "Fuzzy Control", Nikkankogyo Shinbun Publ. Co., 1988.
- [2] L.A. Zadeh, "Fuzzy sets," Inform. Control, Vol. 8, pp. 338-353, 1965.
- [3] C.C. Lee, "Fuzzy Logic in Control Systems: Fuzzy Logic Controller-Part I, II", IEEE Trans. on SMC Vol. 20, No. 2, pp. 404-435, 1990.
- [4] 변중남, "애매논리를 이용한 자동제어 기술", 한국과학기술원 단기강좌 매뉴얼, 1990.