

# 자기 동조 퍼지 논리 제어를 위한 학습 알고리즘의 성능분석

正會員 龔 辰 鉉\* 正會員 李 鎮 赫\*

## Performance analysis of learning algorithm for a self-tuning fuzzy logic controller

Chin Hyun Chung\*, Jin Hyuk Lee\* *Regular Members*

### 요 약

본 논문에서는 퍼지 제어 시스템에 사용되는 퍼지 논리 제어기의 성능을 향상시키기 위한 여러가지 알고리즘들 중에서 학습기법에 속하는 퍼지 메타 규칙에 기초한 자기 동조 기법을 사용하여 직류 서보 전동기 제어를 위한 자기 동조 퍼지 논리 제어를 구현해서, 자기 동조 퍼지 논리 제어기의 설계와 시뮬레이션 및 실험 결과를 고찰하고, 그 결과를 일반적인 퍼지 논리 제어기의 결과와 비교하여 자기 동조 퍼지 논리 제어기의 성능을 평가한다.

### ABSTRACT

In this paper, a self-tuning fuzzy logic controller is implemented to control a DC servo motor by the self-tuning technique based on fuzzy meta-rules with learning in several algorithms to improve the performance of the fuzzy logic controller used in a fuzzy control system. Simulations and experimental results of the self-tuning fuzzy logic controller are compared with those of the fuzzy logic controller to evaluate its performance.

\* 광운대학교 제어측공학과 조교수

\*\* 금성사 정보기술 연구소 연구원

論文番號 : 94138

接受日字 : 1994년 5월 20일

### 1. 서론

퍼지 논리 제어기(FLC) [1, 2, 3, 4, 5, 6,]는 특히 고전적인 기법 [7]으로 분석하기가 너무 복잡하거나, 이용 가능한 정보가 정성적으로 또는 부정확하고 불확실하게 해석될 때, 프로세스들은 제어하기 위한 접근법으로 제시되었다. FLC가 유용한 근본적인 이유는 제어될 프로세스의 기초가 되는 동력학을 알지 않아도, 시스템의 정확한 수학적 모델링이 없어도, 경험있는 오퍼레이터가 충분히 프로세스를 제어할 수 있다는 것이다.

많은 경우에 있어서, 퍼지 제어 시스템의 퍼지 규칙의 집합은 요구되는 수준의 성능에 도달할 때까지, 시행과 오차를 통하여 동조되는 것이 필요하다. 이러한 FLC의 제한은 플랜트의 동작조건의 변화를 예측하고 조정하는 것을 어렵게 함으로, 이러한 어려움을 극복하기 위하여 FLC의 학습에 의하여 제어기의 성능을 향상시킬 수 있는 지능 제어기를 개발하고 프로세스의 동력의 변화에 응답을 적용시키는 것이 요구된다.

이러한 퍼지 제어 시스템에 사용되는 퍼지 논리 제어기의 성능을 향상시키기 위하여 여러가지 알고리즘 [1, 6, 8, 9, 10, 11, 12, 13, 14] 이 제시되었는데, 이러한 알고리즘은 신경회로망 이론과 인공지능에서 나온 적응기법과 학습기법을 사용한 것이다.

자기 동조 퍼지 논리 제어기 [1, 6, 8, 9, 11] 는 새로운 동작 환경에 스스로 적용할 수 있을 뿐만 아니라,

기준입력과 출력을 비교하여 제어 목표에 도달하도록 과거의 규칙을 스스로 개선함으로써, 퍼지 제어 규칙을 개발하는 일을 훨씬 더 간단하게 하고, 단지 설계자에게 대체로 정확한 초기 규칙의 설정만을 요구한다. 그러므로, 설계자에게 제어 규칙을 직접 동조시키는 부담을 덜어준다.

본 논문에서는 반복 학습과 실시간 학습으로 이루어진 퍼지 메타 규칙 [8]에 기초한 자기 동조 기법을 사용하여, 직류 서보 전동기에 결정적인 부하 변동이 생겼을 때, 기준입력을 추종하는 전동기 속도 제어를 위한 자기 동조 퍼지 논리 제어기를 구현한다. 그리고, 자기 동조되지 않은 일반적인 퍼지 논리 제어기와 자기 동조 퍼지 논리 제어기의 성능을 컴퓨터 시뮬레이션을 통하여 비교 평가하고, 실제 실험 결과와 시뮬레이션의 결과를 비교 고찰한다.

### II. 퍼지 논리 제어기

FLC는 전문가의 제어지식을 언어적 형태로 기술한 제어 규칙을 제어기가 동작하도록 역할을 바꾸어 주는 기능을 하는 것이다. 그림 1은 일반적인 퍼지 논리 제어기의 구조를 보여주고 있는데, 보통 퍼지화기, 지식 베이스, 퍼지 추론기, 비퍼지화기의 4부분으로 구성된다.

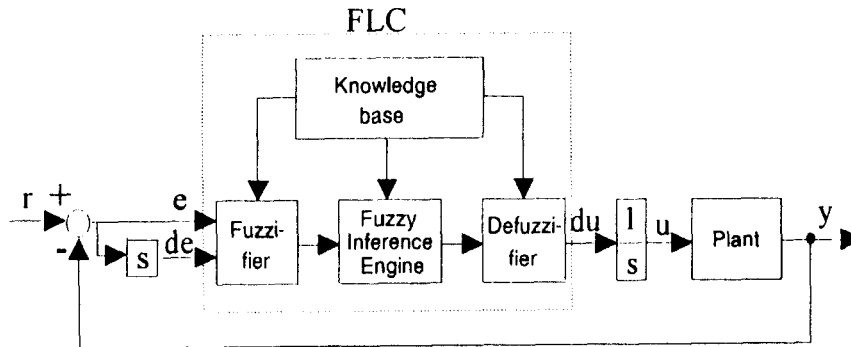


그림 1. 퍼지 논리 제어기의 일반적인 구조

Fig 1. General structure of a fuzzy controller

- $r$  = 기준입력(Reference),  $y$  = 출력(Output),  $e$  = 오차(Error)  $= r - y_k$
- $de$  = 오차의 변화분(Differential of error)  $= e_k - e_{k-1} = y_{k-1} - y_k$
- $u$  = 제어입력(Control input)
- $du$  = 제어입력의 변화분(Differential of control input)  $= U_k - U_{k-1}$

2.1. 퍼지화기

퍼지 제어기는 퍼지 논리를 응용함으로 하나의 명확한 값(Crisp Value)으로 측정된 소거된 입력 변수 값도 적당한 퍼지값(퍼지집합)으로 바뀌어져야 하는데, 이 작업은 퍼지화기에서 이루어진다. 즉, 기존 집합에서 얻어진 값을 퍼지 논리 제어기의 추론 과정에서 사용할 수 있도록 퍼지집합의 영역으로 바꾸어 주는 과정이다.

2.2. 지식 베이스(Knowledge Base)

지식 베이스에는 제어 대상에 대한 지식과 제어목적이 기술된다. 지식 베이스는 데이터 베이스와 퍼지 규칙 베이스의 두 부분으로 이루어진다.

2.2.1. 데이터 베이스

데이터 베이스는 퍼지 논리 제어기의 제어 규칙의 작성과 적용 및 퍼지 추론에 기본이 되는 다음과 같은 3가지 사항을 정의해 준다.

- a. 입력변수의 전체집합의 이산화 및 정규화 방식
- b. 퍼지집합의 멤버쉽 함수 형태
- c. 입·출력 공간의 분할 방식

2.2.2. 퍼지 규칙 베이스

퍼지 규칙 베이스에는 제어 대상으로부터 발생하는 일련의 동작을 조절하는 "IF (조건부) THEN (결론부)" 형식의 언어적 규칙으로 표현된다. 일반적인 규칙의 형태는 다음과 같다.

Rule: "IF x is A and y is B THEN z is C".

여기서 x, y는 제어기 입력 정보로 사용된 시스템의 상태변수이고, z는 제어 입력변수이며, A, B, C는 전체 집합 U, V, W에서 정의된 x, y, z의 퍼지값(퍼지집합)이다.

2.3 퍼지 추론기

이 부분은 퍼지 논리 제어기에서 가장 중요한 부분으로 퍼지 규칙 베이스에서 사용된 퍼지 제어 규칙을 기초로 하여 결론을 추론하는 논리 연산 부분이다. 다음과 같이 두 개의 입력변수와 하나의 출력변수를 갖는 두 개의 퍼지 규칙으로 이루어졌다고 하고 Mamdani의 Min-Max연산자를 사용한 경우의 추론 과정은 그림 2와 같다.

Rule1 : IF x is A1 and y is B1 THEN z is C1

Rule2 : IF x is A2 and y is B2 THEN z is C2

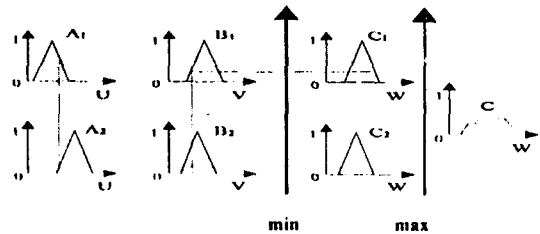


그림 2. Mamdani의 퍼지 추론 방법

Fig 2. Mamdani's fuzzy inference method

2.4. 비퍼지화기

퍼지 제어의 퍼지 추론의 결과 출력변수는 단일 실수값이 아닌 퍼지집합으로 출력된다. 따라서, 단일한 실수값을 얻는 알고리즘이 필요하다. 본 논문에서는 가장 널리 쓰이는 방법인 무게 중심법을 사용하였다. 무게 중심법(COG)을 취하는 식은 아래와 같다.

$$Z^* = \frac{\sum_{j=1}^n Z_j \cdot \mu_c(Z_j)}{\sum_{j=1}^n \mu_c(Z_j)}$$

여기서, n은 제어변수 전체집합의 이산화 준위 또는 등급(Quantization Level).

II. 자기 동조 퍼지 논리 제어기와 제어 알고리즘

3.1. 자기 동조 퍼지 논리 제어기

자기 동조 퍼지 논리 제어기(STFLC)는 기준 입력의 변화나 외란에 의하여 동작 환경이 변하여도 스스로 시스템 파라미터를 최적의 값으로 동조시켜 새로운 동작 환경에 적용한다. 기준입력과 출력을 비교하여 제어 목표에 도달하도록 과거의 규칙을 스스로 개선하여 퍼지 제어 규칙을 개발하는 일을 손쉽게 하여, 설계자가 직접 제어 규칙을 동조시킬 필요가 없다.

본 논문에서 제시한 자기 동조 퍼지 논리 제어기의 구조는 그림 3과 같으며, 자기 동조 퍼지 논리 제어기는 반복 학습과 실시간 학습으로 이루어진 퍼지 메타 규칙에 의해서 스케일링 계수를 동조시키고 규칙을 향상시켜서, 요구되는 성능에 도달할 때까지 제어결과를 동조시킨다. 또한, 퍼지 메타 규칙은 동조에 사용되는

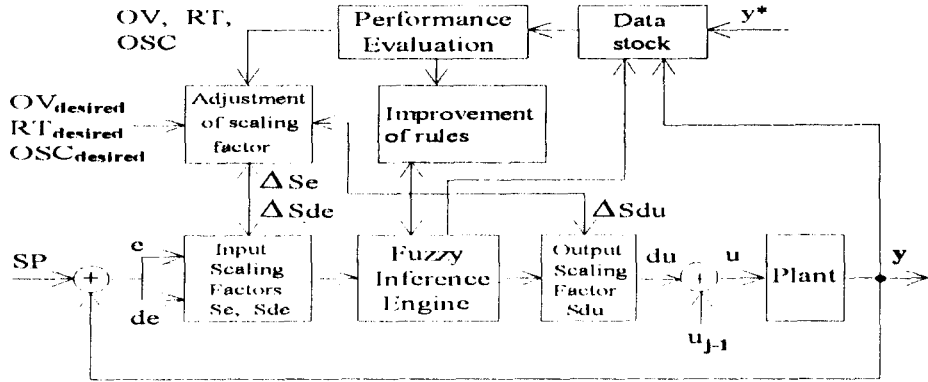


그림 3. 자기 동조 퍼지 논리 제어기의 구조

Fig 3. Structure of a self tuning fuzzy logic controller

규칙들 중에서 적절한 영역만을 선택하기 때문에, 동조 시 사용되는 규칙의 수가 적어 보다 빠른 제어기의 응답특성을 기대할 수 있다. 여기서 자기 동조 퍼지 논리 제어기는 두 개의 제어입력 변수, 속도 오차  $e$ 와 그의 변화분  $de$ 를 가진다. 제어기에 의해서 생성된 제어동작은 조작량  $u$ 의 변화분  $du$ 이다. 그리고 SP(Set Point)는 기준 입력을 의미한다.

두 제어입력 변수  $e$ 와  $de$  출력 변수  $du$ 에 대한 스케일링 계수  $Se, Sde, Sdu$ 의 동조는 퍼지 메타 규칙의 집합에 의해 자동적으로 동조된다. 여기서  $Se, Sde, Sdu$ 는  $e$ 와  $de, du$ 의 값에 대한 가중치(weighting value)을 의미한다. 동조에 사용된 성능척도는 플랜트 과도응답의 오버슈트(Overshoot)와 도착시간(Reaching

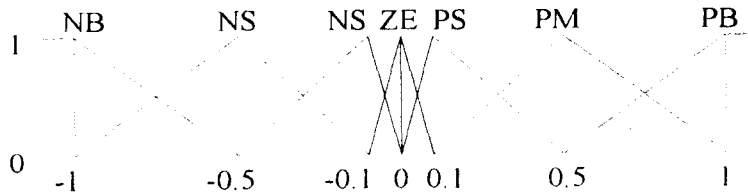


그림 4. 제어규칙에서 사용된 퍼지집합의 멤버쉽 함수

Fig 4. Membership functions of fuzzy sets used in the control rules

그림 4와 표 1은 각각 퍼지 논리 제어기에서 사용된 멤버쉽 함수와 퍼지 제어 규칙을 보여준다.

Time), 그리고 진동폭(Oscillation) 등이며, 이러한 것은 그림 5에 잘 나타나 있다.

이러한 기법은 Maeda [8]에 의해 제시된 기법에

표 1. FLC에서 사용된 퍼지 제어 규칙

Table 1. Fuzzy control rules used in the FLC

$de \backslash e$	NB	NM	NS	ZE	PS	PM	PB
NB	NB	NB	NB	NB	NM	NS	ZE
NM	NB	NB	NM	NM	NS	ZE	PS
NS	NB	NM	NM	NS	ZE	PS	PM
ZE	NB	NM	NS	ZE	PS	PM	PB
PS	NM	NS	ZE	PS	PM	PM	PB
PM	NS	ZE	PS	PM	PM	PB	PB
PB	ZE	PS	PM	PB	PB	PB	PB

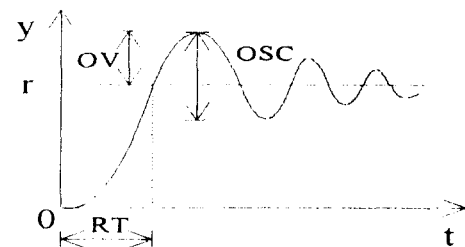


그림 5. 계단응답의 성능척도

Fig 5. Performance measures of a step response

기초하는데, 퍼지 논리 제어기의 입출력에 적용된 입력과 출력 스케일링 계수들은 FLC의 전체 성능에 의하여 요구되는 응답을 얻기 위해 동조된다.

### 3.2 자기 동조 알고리즘

#### 3.2.1 스케일링 계수의 조정

제어가 끝났을 때, 자기 동조 퍼지 논리 제어기의 반복 학습 기능은 제어결과를 평가하여 스케일링 계수를 조정한다.

동조에 사용된 규칙은 다음과 같다.

- IF performance-measure is X1, THEN  $\Delta Se$  is Y1.
- IF performance-measure is X2, THEN  $\Delta Sde$  is Y2.
- IF performance-measure is X3, THEN  $\Delta Sdu$  is Y3.

한편, 성능척도는 오버슈트(OV)와 도착시간(RT), 그리고 진동폭(OSC)중 하나이어야만 한다.  $X_1$ 과  $X_2$ ,  $X_3$ 는 이러한 성능척도를 나타내는 퍼지집합이고,  $Y_1$ 과  $Y_2$ ,  $Y_3$ 는 스케일링 계수의 수정을 나타내는 퍼지집합이다. 성능척도는 목표값의 퍼센트 오버슈트( $OV_{desired}$ ), 도착시간( $RT_{desired}$ ), 퍼센트 진동폭( $OSC_{desired}$ )에 의해서 얻어진다. OV, RT, OSC는 실제 얻어진 값이며,  $OV_{desired}$ 와  $RT_{desired}$  그리고  $OSC_{desired}$ 는 요구되는 값이다. 그러므로, 실제 성능척도는 다음과 같다.

$$\begin{aligned} \Delta OV &= OV - OV_{desired} \\ \Delta RT &= RT - RT_{desired} \\ \Delta OSC &= OSC - OSC_{desired} \end{aligned}$$

이러한 평가치는 제어의 결과로 부터 주어지고, 최종성능 FP(Final Performance)에 사용된다.

$$FP \cong \min(\mu_{OV}^{(\Delta OV)}, \mu_{RT}^{(\Delta RT)}, \mu_{OSC}^{(\Delta OSC)}), \quad 0 \leq FP \leq 1.$$

위의 식이 만족되어질 때, 조정이 끝나게 된다. 그리고 최종성능 FP는 0에서 1사이의 값을 설정할 수 있으며, 에 가까운 값을 설정할수록 더 좋은 제어응답을 얻을 수 있는데, 이 값의 설정은 제어기 설계자가 어느 정도의 성능을 원하느냐에 따라 설정치를 정하게된다.

스케일링 계수의 동조에 사용된 규칙과 멤버십 함수는 표 2와 그림 6과 같다. 각각 동조절차를 반복한 후에, 스케일링 계수 Se와 Sde, Sdu는 아래와 같이 새로워진다.

$$Se(i+1) = Se(i) + (1-FP)\Delta Se$$

$$\begin{aligned} Sde(i+1) &= Sde(i) + (1-FP)\Delta Sde \\ Sdu(i+1) &= Sdu(i) + (1-FP)\Delta Sdu \end{aligned}$$

이러한 절차는 요구되는 성능에 도달할 때까지 반복된다.

표 2. 스케일링 계수의 동조에 사용된 규칙  
Table 2. Rules used for tuning scale factors

Antecedent		Consequent		
Performance		$\Delta Se$	$\Delta Sde$	$\Delta Sdu$
$\Delta OV$	$N_1$	NS <sub>11</sub>	PM <sub>12</sub>	PB <sub>13</sub>
	$P_1$	PB <sub>11</sub>	NM <sub>12</sub>	NS <sub>13</sub>
$\Delta RT$	$N_2$	PB <sub>21</sub>	NB <sub>22</sub>	NB <sub>23</sub>
	$P_2$	NB <sub>21</sub>	PB <sub>22</sub>	PB <sub>23</sub>
$\Delta OSC$	$P_3$	PB <sub>31</sub>	NB <sub>32</sub>	NB <sub>33</sub>

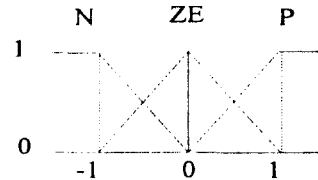


그림 6. 스케일링 계수를 동조시키기위해 사용된 성능 척도 OV, RT, OSC를 나타내는 퍼지집합의 멤버십 함수

Fig 6. Membership functions of fuzzy sets describing performance measures OV, RT, OSC used for tuning scale factors

#### 3.2.2 규칙의 항상

먼저, 그림 7에서 보여진 것처럼 목표응답을 결정한다. 목표응답은 목표치(r), 목표 도착시간( $RT^*$ ), 지연시간(L)에 의해 결정된다. 이러한 기준은 각 샘플링 시간에 목표응답과 제어응답이 어느 정도 일치하는지를 평가하는데 사용된다.

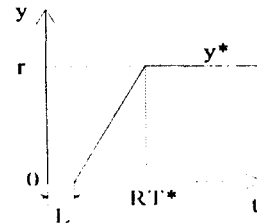


그림 7. 목표응답

Fig 7. The target response

제어 규칙의 향상을 위한 실시간 학습방법은 과거에 사용된 규칙을 조정하는 것이다. 그리고, 제어응답이 각 샘플링 포인트에서 얻어졌을 때 제어응답이 목표 응답과 일치하도록 과거에 사용된 규칙들은 조정된다.

이제, 제어규칙의 결론부는 실수  $Sd_{ui}(i=1\sim 49)$ 를 가진다. 이러한  $Sd_{ui}$ 는 각 샘플링 포인트에서 조정된다.

If  $e_k^*$  is  $X_1$ ,  $de_k^*$  is  $X_2$  THEN  $\Delta Sd_{ui}$  is  $Y_1$

$$e_k^* = y_k^* - y_k, de_k^* = e_k^* - e_{k-1}^*$$

$\Delta Sd_{ui}$ : 규칙 i에서 파라미터  $Sd_{ui}$ 의 조정치

$e_k^*$ : 응답의 오차,  $de_k^*$ :  $e_k^*$ 의 오차

$y_k^*$ : 목표응답,  $y_k$ : 제어응답

위의 규칙에서 퍼지집합  $X_1, X_2, Y_1$ 은 그림 4의 멤버십 함수와 표1의 제어 규칙을 사용한다. 그리고, 새로운  $Sd_{ui}$ 는 아래의 식에 의해서 계산된다.

$$Sd_{ui}^{NEW} = Sd_{ui}^{OLD} + \Delta Sd_{ui} \cdot W_i^{(k-m)}$$

$W_i^{(k-m)}$ : 샘플링 시간 (k-m)에서 규칙의 적용도

$Sd_{ui}^{OLD}$ : 샘플링 시간 (k-m)에서 사용된 규칙들의 결론부의 값

$Sd_{ui}^{NEW}$ :  $Sd_{ui}^{OLD}$ 의 향상된 값

응답은 제어의 결말에 평가되고, 최종성능 IP를 구하는 공식이 만족되었을 때 조정은 끝난다.

#### IV. 시뮬레이션

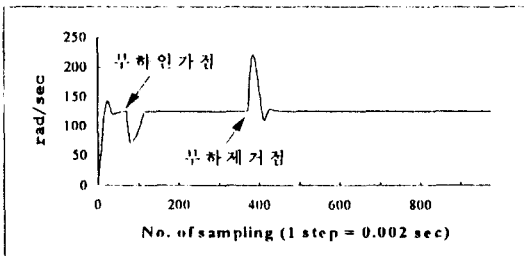


그림 8. 일정속도 기준입력시 순간적인 부하 변동에 대한 FLC의 응답

Fig 8. Response of FLC to a temporary load variation in the case of constant reference

본 논문에서 제시한 자기 동조 제어 알고리즘의 우수함을 보이기 위해서 자기 동조 제어 알고리즘을 적용한 FLC와 동조되지 않은 일반적인 FLC의 제어 알고리즘을 직류 서보 전동기 시스템에 결정적인 부하 변동이 생겼을 때 각 알고리즘을 적용한 제어기의 기준속도 추종상태를 조사하여 각 제어기의 성능을 평가한다. 본 논문에서는 동일한 2개의 100W 직류 서보 전동기를 커플링 시킨 모델을 대상 시스템으로 선정하였다.

그리고, 시뮬레이션 해석 조건은,

1. 125.7  $\frac{rad}{sec}$  (1200rpm)의 기준입력을 인가한다.
2. 2초간 단방향 회전시, 순간적인 부하( $R_L=5\Omega$ )를 인가한다. (70 step에서 부하인가, 380 step에서 부하제거). 한편, 샘플링 시간은 0.002초이다.
3. 1초동안 정방향 회전후 1초동안 역방향 회전을 8회 반복 실행한다. 한편, 샘플링 시간은 0.004초이다. 1100 step이후 지속적인 부하 변동시 기준속도를 추종해가는 것을 조사한다.
4. 먼저, 일정속도에 대한 순간적인 부하변동과 기준입력시 지속적인 부하 변동에 대한 각 제어기의 응답을 조사한다.

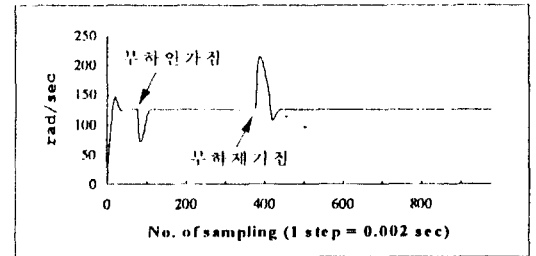


그림 9. 일정속도 기준입력시 순간적인 부하 변동에 대한 STFCLC의 응답

Fig 9. Response of STFCLC to a temporary load variation in the case of constant reference

먼저, 그림 8, 9와 같이 일정속도 기준입력시 순간적인 부하변동에 대한 STFCLC와 FLC 두 제어기에 대한 성능을 평가하면, STFCLC가 FLC에 비해서 초기 과도상태의 도착시간과 진동폭이 작으며, 순간적인 부하 변동 (70 step에서 부하인가, 380 step에서 부하제거)에 대해서도 STFCLC가 FLC에 비해서 정상상태에 더 빨리 도달하는 것 즉, 회복능력이 우수하다는 것을 알 수

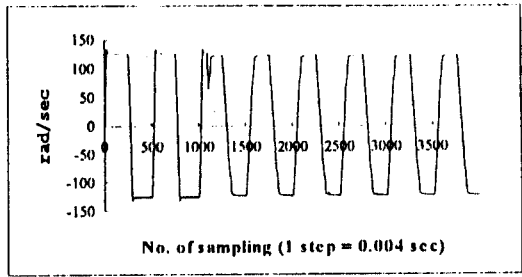


그림 10. 구형과 기준입력시 지속적인 부하 변동에 대한 FLC의 응답

fig 10. Response of FLC to a continuous load variation in the case of square wave reference

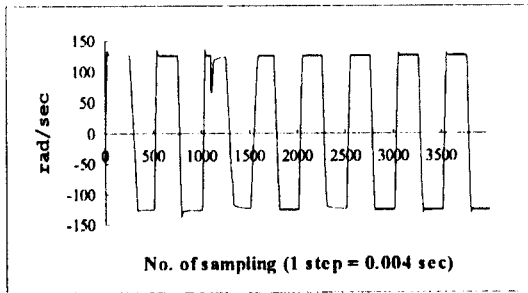


그림 11. 구형과 기준입력시 지속적인 부하 변동에 대한 STFLC의 응답

fig 11. Response of STFLC to a continuous load variation in the case of square wave reference

있다.

또한, 그림 10, 11에서 구형과 기준입력의 경우, 지속적인 부하 변동시 (1100 step에서 부하인가)에는 STFLC는 빠른 방향전환(1초간 정방향 회전후 1초간 역방향 회전)에도 불구하고 계속적인 자기 동조에 의해서 스케일링 계수를 변화시키고 제어 규칙을 향상시켜서 기준속도를 빠르게 추종해가지만, FLC는 빠른 방향 전환에 의하여 기준속도를 거의 추종하지 못한다.

위에서 설명한 내용과 같이, 일반적인 퍼지 논리 제어기에 비해서 자기 동조 퍼지 논리 제어기는 퍼지 메타 규칙에 기초한 반복 학습과 실시간 학습에 의해서 스케일링 계수를 동조시키고 제어 규칙을 향상시켜서 제어기의 성능을 향상시킨다.

그러므로, 자기 동조 퍼지 논리 제어기는 일반적인 퍼지 논리 제어기에 비해서 더욱 더 적절한 제어기의 동작 특성을 만들어준다.

## V. 실험

### 5.1 실험장치의 구성

본 논문에서 제시한 자기 동조 퍼지 논리 제어기는 C언어를 사용하여 IBM 486PC로 구현하였으며 샘플링 시간은 2 msec이다. 입·출력은 실제로 제작된 인터페이스 보드를 통하여 이루어진다. 데이터는 AT bus의 16bit 데이터 버스를 사용하여 데이터의 고속 전송을 이루었으며, 전동기의 속도는 엔코더에서 출력되는 펄스를 4채배하여 16bit 카운터에서 계수되며, 계수된 펄스의 수는 74LS373에 의해 선택되어 데이터 버스를 통

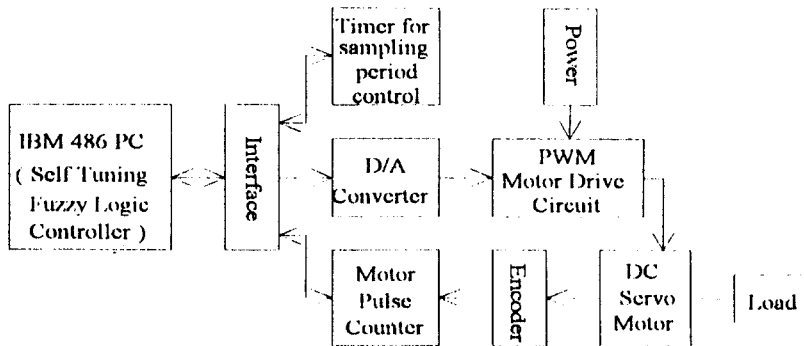


그림 12. 자기 동조 퍼지 논리 제어기와 실험 장치와의 인터페이스

Fig 12. Interface of a self tuning fuzzy logic controller and an experimental unit

하여 PC로 입력된다. D/A 변환기(Digital to Analog Converter)는 12bit의 분해능력을 가진 AD667을 사용하였으며 D/A 변환기의 출력은 모터 구동 및 전력 증폭기의 PWM 회로의 기준입력에 인가되어 모터의 속도를 제어하는 제어신호로 이용된다. 그리고 제어루프에서 샘플링 시간은 8254 PIT(Programmable Inverter Timer)를 이용하여 타이머에 할당된 인터럽트 루틴을 이용하였고 인터럽트 루틴에서 데이터의 입력, 자기 동조 퍼지 논리 제어 알고리즘, 제어신호 출력 등을 실행하였다.

실험은 시뮬레이션과 같이 FLC와 STFCL에 대해서 구형파 기준입력에 대한 순간적인 부하 변동이 발생했을 때의 동작특성을 조사하였다. 부하는 부하측 전동기의 두 출력단자에 저항( $R=5\Omega$ )을 연결하여 실험하였고, 기준속도는 1000 rpm( $104.7 \text{ rad/sec}$ )으로 하였다. 전동기와 부하측 전동기는 동일한 100W 직류 서보 전동기를 이용하였다.

그림 12은 자기 동조 퍼지 논리 제어기와 실험장치와의 인터페이스를 보여주며, 그림 13는 실험 장치도이다.

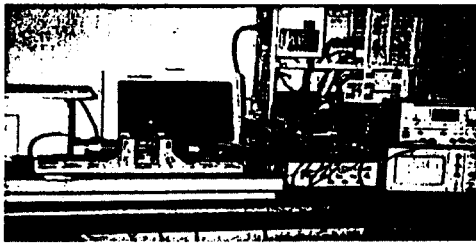


그림 13. 실험 장치도  
Fig 13. Experimental unit

## 5.2 실험결과

실험결과, 두 제어기의 응답은 시뮬레이션과는 다소 차이가 났다. 그 이유는 실험장치의 진동을 확실하게 흡수하지 못했으며, 두 서보 전동기를 커플링하기 위해 사용된 커플러가 정밀하지 못하였기 때문이다. 이로 인하여, 제어기가 적절한 제어입력을 제어시스템에 주어도 제어입력이 적용되는 순간에 위와 같은 오차들에 의해서 실제 시스템에서는 완전히 적합한 제어 입력값이 적용되지 않기 때문이다. 그러나 시뮬레이션과 마찬가지로 STFCL의 응답(그림15)이 FLC의 응답(그림 14)에 비해서 과도응답시 오버슈트와 도착시간에서 성능

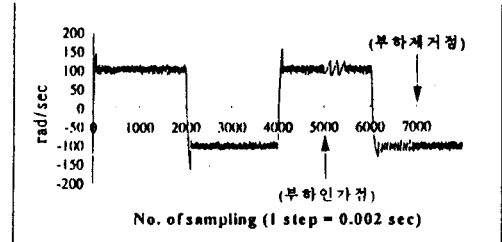


그림 14. 순간적인 부하 변동에 대한 실제 실험을 통한 FLC의 응답

Fig 14. Response of FLC in real experiment to a temporary load variation

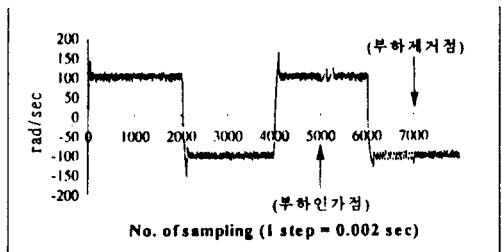


그림 15. 순간적인 부하 변동에 대한 실제 실험을 통한 STFCL의 응답

Fig 15. Response of STFCL in real experiment to a temporary load variation

이 우수하였으며, 부하 변동에 대한 회복 능력면에서는 STFCL의 응답이 FLC의 응답에 비해서 기준속도를 빠르게 추종해 가는 것을 알 수가 있었다.

## VI. 결론 및 추후과제

본 논문에서는 자기 동조 퍼지 논리 제어기의 성능을 평가하기 위해서, 마이크로 프로세서를 이용하여 일반적으로 쉽게 모델링이 가능한 직류 서보 전동기의 속도제어시 부하 변동에 대한 플랜트의 동작 특성을 일반적인 퍼지 논리 제어기에 의한 동작 특성과 비교하여 보았다. 제어 알고리즘으로는 퍼지 메타 규칙에 기초한 반복 학습과 실시간 학습에 의하여 스케일링 계수를 동조시키고 규칙을 향상시켰다.

컴퓨터 시뮬레이션을 통하여 얻은 응답을 분석한 결과, 일반적인 퍼지 논리 제어기에 비해서 본 논문에서 제안된 자기 동조 퍼지 논리 제어기가 순간적인 부하 변동시(그림9) 오버슈트가 작고 도착시간이 빠르며, 정상상태에 빨리 수렴한다. 또한 지속적인 부하 변동시(그림 11)에도 자기 동조 퍼지 논리 제어기는 지속적인



자기 동조에 의해서 빠른 방향전환에도 불구하고 도착 시간이 빠르게 정상상태에 빠르게 수렴하지만, 일반적인 퍼지 논리 제어기는 빠른 방향전환에 의해서 기준 속도를 거의 추종하지 못한다. 이와 같이 본 논문에서 제시한 제어가 수렴속도 및 부하 변동에 대한 회복 능력에서 성능이 우수함을 알 수 있었다.

실험결과, 두 제어기의 응답은 시뮬레이션과는 다소 차이가 났지만, 시뮬레이션과 마찬가지로 STFCL의 응답(그림 15)이 FLC의 응답(그림 14)에 비해서, 부하 변동에 대한 회복 능력면에서 우수하다는 것을 알 수가 있었다.

멤버십 함수와 제어 규칙 베이스를 설계하는 일반적인 규칙은 아직 없으며, 제어할 플랜트에 따라 설계자의 경험과 지식을 바탕으로 시행오차에 의해서 가장 적절한 것을 설정한다. 따라서, 적절한 멤버십 함수와 제어 규칙 베이스를 설정하는 것을 용이하게 하는 방법을 개발하는 것과 실제 실험에 있어서의 오차들을 줄여서 시뮬레이션과 거의 같은 응답을 얻어서 본 논문에서 제시한 자기 동조 논리 제어기의 우수함을 확실하게 보이는 것이 앞으로의 과제로 남는다.

참고 문헌(Reference)

[1] Walter C. Daugherty, Balaji Rathakrishnan, and John Yen, "Performance Evaluation of a Self tuning Fuzzy Controller," IEEE International Conference on Fuzzy Systems, San Diego, pp. 389-397, 1992

[2] C. C. Lee, "Fuzzy Logic in Control Systems, Part I and Part II," IEEE Transactions on Systems, Man and Cybernetics, (2), March, April 1990.

[3] H. K. Berenji, "Fuzzy Logic Controllers," In R. R. Yager and L. A. Zadeh editors, An Introduction to Fuzzy Logic Applications in Intelligent Systems, Kluwer Academic Publishers, pp. 69-96, 1992.

[4] N. K. Alang Rashid and A. Sharif Herger, "Tuning of Fuzzy Logic Controllers by Parameter Estimation Method," In M. Jamshidi, N. Vadiie, and T. J. Ross editors, Fuzzy Logic and Control - Software and Hardware Applications, vol. 2, Prentice-Hall, pp. 374-392, 1993.

[5] S. Yamada, H. Fujikawa, and A. Fuji, "A Tuning Method for a Fuzzy Controller," In Proc. of the international conference on Fuzzy Logic & Neural Networks, pp. 397-400, Iizuka, Japan, July, 1990.

[6] A. Kandel and G. Langholz, "Fuzzy Control Systems," CRC Press, 1994.

[7] P. J. King and E. H. Mamdani, "The Application of Fuzzy Control Systems to Industrial Processes," Automatica, 13 : pp. 235-242, 1977.

[8] Mikio Maeda, Takce Sato, and Shuta Mirakami, "Design of The Self-Tuning Fuzzy Controller," In Proc. of the International Conference on Fuzzy Logic & Neural Networks, pp. 393-396, July 1990. Iizuka, Japan.

[9] M. Maeda and S. Murakami, "Self-Tuning Fuzzy Logic Controller," Trans. of SICE, Japan, 24-2, pp. 191-197, 1988.

[10] M. Yoshida, Y. Tsutsumi, and T. Ishida, "Gain Tuning Method for Design of Fuzzy Control Systems," In Proc. of the International Conference on Fuzzy Logic & Neural Networks, pp. 405-408, Iizuka, Japan, July, 1990.

[11] C-Y Shieh, Satish S. Nair, "A New Self Tuning Fuzzy Controller Design and Experiments," 2nd IEEE International Conference on Fuzzy Systems, San Francisco, pp. 309-314, California, 1993.

[12] Wen-Ruey Hwang, Wiley E. Thompson, "An Improved Method for Designing Fuzzy Controllers for Position Control Systems," 2nd IEEE International Conference on Fuzzy Systems, San Francisco, pp. 1361-1364, California, 1993.

[13] S. Z. He, S. Tan and C. C. Hang, "Control of dynamical processes using an on-line rule adaptive fuzzy control systems," Fuzzy Sets and Systems 54, 1993.

[14] 서영노, 정진현, "자기학습 구조를 가진 퍼지 제어기의 응용," 한국통신학회 논문지, Vol. 19, No.6, pp, 1182-1189, June 1994.

龔辰鉉 (Chin Hyun Chung) 정회원

第19卷 第6號 參照

현재 : 광운대학교 제어계측공학과 조교수



李鎭赫 (Jin Hyuk Lee) 정회원

1968년 3월 5일생

1992년 2월 : 광운대학교 제어계측  
공학과(공학사)

1994년 8월 : 광운대학교 제어계측  
공학과(공학석사)

1994년 7월 ~ 현재 : 금성사 정보기  
술연구소 연구원