

PID 와 자동 학습 퍼지 제어기를 이용한 도립 진자의 제어

*이 용 노, *김 태 원, *서일홍, **김기엽
* 한양대학교 전자공학과, ** 한국기계연구소 자동화 연구부

A Novel Self-Organizing Fuzzy Plus PID Type Controller with Application to Inverted Pendulum Control

*Yong-No Lee, *Tae-Won Kim, *Il-Hong Suh, **Gi-Yeop Kim
*Dept. of Electronic Eng. Hanyang Univ., **Korea Institute of Machinery & Metals

ABSTRACT

In this paper, a novel self-organizing fuzzy plus PID control algorithm is proposed and analyzed by extensive computer simulations and experiments with an inverted pendulum. Specifically, the proposed self-organizing fuzzy controller consists of a typical fuzzy reasoning part and self organizing part in which both on-line and off-line algorithms are employed to modify the 'then' part of the fuzzy rules and to decide how much fuzzy rules are to be modified after evaluating the control performance, respectively. And the fuzzy controller is replaced by a PID controller in a prespecified region near by the set point for good settling actions.

1. 서론

1969년 퍼지 집합론[1]이 발표된 이후 각 분야에서 이를 응용한 사례들이 발표되고 있다[2][3]. 이중 퍼지 로직 컨트롤(Fuzzy Logic Control ; FLC)은 가장 활발하고 엔지니어들에게 관심을 끄는 분야가 되었고, FLC의 일반적인 구조와 제어 규칙의 구성 방법, 추론 방법들이 제시되었다[4-6]. FLC가 제어 엔지니어들의 관심을 끄는 이유는 FLC 알고리즘이 경험, 직관 등에 의한 주먹 구구식의 규칙을 언어 변수로 표현하고 구현할 수 있으며 제어 대상 시스템의 수학적 모델링이 필요하지 않다는 점 때문이다[6]. 특히, 실시간 제어를 하기 위해서 모든 입력 변수와 출력값을 정량화하고 이들을 이용 미리 추론을 하여 룩업 테이블(Look-Up Table ; LUT)로 만들어 두는 방법이 제안되었다[6][14]. 그러나 FLC는 많은 장점에도 불구하고 제어 규칙을 반복적인 실험을 통해 만족할 만한 성능이 나올때 까지 시행착오적으로 수정 해야한다는 단점을 가지고 있다. 비록 체계적인 제어 규칙의 수정 방법들이 제안되었지만 이런 작업들은 지루한 일이다. Mamdani 와 Procyk[7]은 반복 학습을 통해 제어 규칙을 수정하는 자동 학습 제어기(Linguistic Self-Organizing Process)를 처음으로 제안하였고, 뒤를이어 Yamazaki, Sugeno[8], Shao[9], Maeda, Murakami[10]등이 자동 학습 제어기에 관한 연구 결과를 발표하였으며, Tansheit[11]은 이를 로보틱스 분야에 적용하였다. [7-11]에서는 퍼지 제어기의 성능 평가 방법으로 매 샘플링 순간에서의 오차와 오차의 변화량을 사용하였으며 오차가 큰 구간에서는 제어 입력을 크게하고 오차가 작고

빠른 속도로 목표 지점에 접근하는 구간에서는 제어 입력을 작게 하는 학습 알고리즘을 이용하였다. 그러나 규칙의 수렴 속도를 높이면 제어 규칙이 일정구간 내에서 진동하게 되고, 제어 규칙을 점근적으로 수렴하게 하면 수렴 속도가 늦어지는 현상이 발생한다. [10]에서는 제어 규칙을 수정하지 않고 온라인(On-Line)으로 입력 이득 상수를 조정하고 오프라인(Off-Line)으로 출력 이득 상수를 조정하는 알고리즘을 이용하였다. 그러나 출력 이득 상수가 수렴하지 않고 계속 변화하는 현상을 보인다.

본 논문에서는 제어 규칙이 점근적으로 수렴하면서도 수렴 속도를 향상 시킬수 있는 자동 학습 퍼지 제어기(SOFC)를 제안하고 이를 도립 진자를 대상으로한 시뮬레이션과 실험을 통해 성능이 개선됨을 보인다. 그러나 초기 제어 규칙에 따라 대부분의 SOFC가 그렇듯이 학습이 되지않을 수가 있다. 따라서 자동 학습 이전 상태의 FLC 이득 상수들을 경험적 방법에 의해 조심스럽게 결정하여야 한다. Braae 및 Rutherford는 [12][13]에서 제어 규칙의 상태도위에 제어 대상 시스템의 출력 궤적(Trajectory)을 도시하고 이를 분석하여 바람직한 출력 궤적을 형성하도록 제어 규칙과 제어기의 이득 상수를 조정하는 방법을 제시하였다. 그렇지만, 제어 규칙의 상태도 위에 제어 대상 시스템의 출력 궤적을 도시하기란 애매하다. 왜냐하면 퍼지 제어기의 출력값은 제어 규칙중의 어느 하나에 의해서만 결정되는 것이 아니고 여러 규칙이 서로 밀접하게 연관되어 출력을 형성하기 때문이다.

본 논문에서는 보다 명확하게 제어 대상 시스템의 출력 특성을 분석하기 위하여 LUT위에 출력 궤적을 도시하고 이를 분석하여 FLC의 이득 상수들을 결정하는 경험적 방법을 제시한다.

한편 FLC는 노이즈나 제어 대상 시스템의 파라메타의 불확실성 또는 변화에 둔감하다는 장점이 있으나 목표 지점 근처에서의 정밀 제어가 어렵다는 단점이 있다. 이를 보상하기 위하여 Li 와 Lau[14]는 제어 규칙과 언어 변수값을 정밀 제어를 위한 것과 도달 시간을 빠르게 하기 위한것의 두가지로 나누었으며, Yoshida[15]는 좌우 대칭이 아니고 ZE(Zero)에 접근할수록 폭이 점점 좁아지는 형태의 언어 변수값을 제안하였다. 그러나 이 두 경우 모두 언어 변수의 폭을 ZE부근에서 좁게하여 분해능(Resolution)을 개선했지만 노이즈에 민감하게 되었다. 더구나 제어 입력들이 룩업 테이블화 되어있기 때문에 연속적이지 않고 단속적이다. 따라서 목표지점 부근에서 제어 입력이 진동하여 정밀 제어가 어려워지는 결과를 초래한다.

본 논문에서는 이러한 현상을 제거하기 위하여 단속적 제어 입력을 생성하는 룩업 테이블 형태의 FLC 대신에 목표 출력값

부근에서는 연속적인 제어 입력을 생성시키는 PID 알고리즘을 이용하여 정밀 제어를 하고자 한다.

내용중의 명확한 수식 표기를 위하여 퍼지 집합은 영문 대문자로 표기하고 일반 집합(Crisp Set)은 소문자로 표기한다. 또한 언어 변수 (E, \tilde{E} , U)는 첨자없이 표기하고 언어 변수값(E_i , \tilde{E}_i , U_i)은 첨자를 붙이며 어떤 시간에서의 언어 변수의 값 ($E(k)$, $\tilde{E}(k)$, $U(k)$)은 괄호 안에 시간을 표시한다. 또한 \bar{A} 는 A의 여집합을 의미한다.

2. 자동 학습 퍼지 제어기(SOFC)의 설계

FLC는 제어 규칙을 반복적인 실험을 통해 만족할 만한 성능이 나올때 까지 시행 착오적인 시도 및 수정 과정을 통해 설계를 해야한다. 비록 체계적인 제어 규칙의 수정 방법들이 제안되고는 있지만[12][13] 이런 작업들은 지루한 일이다. 따라서 시스템이 원하는 출력을 발휘할수 있도록 제어 규칙을 스스로 조정하는 제어기가 요구된다. 이를 위하여 본 절에서는 새로운 자동 학습 제어기를 제안한다.

자동 학습 제어기의 새로운 구조는 그림1에서 보는바와 같이 일반적 FLC로 구성되는 하위층과 제어 규칙의 조정을 담당하는 상위층으로 구성된다. 먼저 본 논문에서 사용하고자 하는 하위층 FLC의 제어 규칙은 입력 퍼지 변수로써 오차 E 및 오차의 시간에 따른 변화량 \tilde{E} 를 이용하고 출력 퍼지 변수로써 U를 선택 할때 입력과 출력의 관계(Relation)가 IF THEN ELSE(IF E is E_i and \tilde{E} is \tilde{E}_i THEN U is U_i ELSE ...)의 형태로 구성된 것이며, i번째 규칙의 입출력 관계 R_i 는 [7]에서와 같이 (1)식의 카테이션 프로덕트(Cartesian Product)로 표현된다.

$$R_i = E_i \times \tilde{E}_i \times U_i \quad (1)$$

여기서 E_i , \tilde{E}_i , U_i 는 언어 변수 E, \tilde{E} , U의 값(Linguistic Value)이다. 이제 k번째 샘플링 시간에서 $U(k)$ 는

$$U(k) = \{ E(k) \times \tilde{E}(k) \} \circ R \quad (2)$$

으로 구하며 여기서 \circ 는 Zadeh[4]의 합성 규칙 (Composition Rule of Inference)인 Sup-Min연산을 이용한다. 한편, 실제 시스템에서의 제어 입력은 퍼지량이 아닌 일정한 값(Crisp Value)이므로 비 퍼지화(Defuzzification) 과정이 필요하다. 비 퍼지화의 방법으로는 다음의 면적 중심법[6]을 이용한다.

$$u(k) = \left\{ \sum_n u_n \cdot \mu U(k)(u_n) / \sum_n \mu U(k)(u_n) \right\} \quad (3)$$

또한 실시간 제어를 위한 LUT를 사용하기 위하여 정량화 과정을 도입한다. 물리량 p를 q+1 단계로 정량화 시키는 함수 $\psi(\cdot)$ 는

$$\psi(p) = \text{Min} [q, \text{Max} \{0, [\Phi \{ (p - p_{\min}) / \text{Step} \} + 1]] \} \quad (4)$$

이며, 여기서 $\text{Step} = (p_{\max} - p_{\min}) / (q-1)$, $\Phi(\cdot)$ 는 절단 (Truncation) 함수이다. 특히 q가 짝수이고 0을 중심으로 좌우 대칭이 되도록 정량화 시키는 함수 $\psi_{sy}(p)$ 는

$$\psi_{sy}(p) = \psi(p) - q/2 \quad (5)$$

이다. 정량화의 과정에서 p_{\min} 과 p_{\max} 의 결정은 시스템의 특성을 미리 파악하여 결정 해야 한다. 이제 (5)식으로 정량화된 입력 변수들을 이용 (2)식의 추론 과정과 (3)식의 비퍼지화 과정을 거쳐 LUT가 만들어진다[14]. 결국 $t = kT$ 에서 FLC의 출력 $u(k)$ 는

$$u(k) = T \{ \psi_{sy}[\alpha_2 \cdot \dot{e}(k)], \psi_{sy}[\alpha_1 \cdot e(k)] \} \quad (6)$$

이다. 이때 $T(\dot{e}, e)$ 는 오차와 오차의 변화량에 대한 정량화 단계가 e 및 \dot{e} 인 경우에 LUT의 출력이다.

이제 상위층의 자동 학습부를 제시하기 위해 먼저[7-9][11] 등에서 제시한 방법과 문제점을 간략히 보기로 하자. 이들 방법은 다음식과 같이 제어 규칙을 수정하도록 구성되어 있다.

$$R(k+1) = \{ R(k) \cap R_i(k) \} \cup R'_i(k) \quad (7)$$

$$R'_i = E(k-1) \times \tilde{E}(k-1) \times U'(k-1) \quad (8)$$

$$U'(k-1) = U(k-1) + U^*(k-1) \quad (9)$$

$$U^*(k-1) = \{ E(k) \times \tilde{E}(k) \} \circ R_p \quad (10)$$

여기서 R_p 는 제어 규칙을 수정하는 규칙이며 이것은 오차가 큰 구간에서는 FLC 제어 규칙의 출력을 더 크게하고 오차가 작고 빠른 속도로 목표 지점에 접근하는 구간에서는 출력을 작게 만들어 주도록 구성되어 있다. 즉 현재의 오차와 시간에 따른 오차의 변화량 $e(k)$ 및 $\dot{e}(k)$ 이 발생한 이유는 바로 이전 샘플링 시간에서 가해진 제어 입력 - 지연 시간이 있는 시스템의 경우는 지연 시간 이전의 제어 입력 - 이 잘못 결정되었기 때문이므로 바로 이전 샘플링 시간에 가해진 제어 입력을 수정해야 하며 이를 위해서는 제어 규칙을 수정하여야 한다는 알고리즘으로 구성되어 있다. 그러나 제어 규칙을 수정하는 규칙 역시 고정되어 있다. 따라서 수정하는 최소량을 크게하면 규칙의 수렴 속도는 향상되지만 제어 규칙이 일정 구간 내에서 진동하게 되고, 작게하면 규칙이 수렴하지만 수렴 속도가 늦어지는 현상이 발생한다. 또한 제어 규칙을 수정할 경우 매 샘플링 시간마다 추론을 다시해야 하므로 실시간 제어에 어려움이 따른다. 따라서 실시간 제어를 위해서는 제어 규칙을 수정하는 것보다 LUT의 출력 값을 수정해야 한다. 또한 제어 규칙의 수정량을 가변시키는 방법을 이용해야 한다. 따라서 본 논문에서는 그림1의 상위층을 온라인(On-Line)으로 룩업 테이블을 수정하는 제어 규칙 수정기와 오프라인(Off-Line)으로 수렴 속도 향상을 위해 제어 규칙 수정기의 출력 이득 상수 λ 를 조정하는 이득 상수 조정기로 구성하고자 한다.

먼저 정량화와 추론 과정을 거쳐 만들어진 룩업 테이블은 다음과 같은 수정 방법을 이용 온라인(On-Line)으로 수정한다.

$$T'(\cdot) = T(k)(\cdot) + \Delta u(k) \cdot \lambda \quad (11)$$

$$T(\cdot; k+1) = \{ T(\cdot; k) \cap \bar{T} \{ \psi_{sy}[\alpha_2 \cdot \dot{e}(k-1)], \psi_{sy}[\alpha_1 \cdot e(k-1)] : k \} \} \cup T' \{ \psi_{sy}[\alpha_2 \cdot \dot{e}(k-1)], \psi_{sy}[\alpha_1 \cdot e(k-1)] : k \} \quad (12)$$

여기서 $T(\dot{e}, e; k)$ 는 $t = kT$ 에서 정량화 단계 \dot{e}, e 에 해당하는 LUT의 출력값이다. 즉 현재 LUT에서 바로 이전 시간에 사용한 출력 부분을 자동 학습 규칙에 의해 수정된 값으로 교체 시키는 것을 의미한다. 이때 수정해야할 양 $\Delta u(k)$ 은 다음 식과 같이 구한다.

$$\Delta u(k) = \sum_n u_n \cdot \mu_{\Delta u(k)}(u_n) / \sum_n \mu_{\Delta u(k)}(u_n) \quad (13)$$

$$\Delta U(k) = \{ E_p(k) \times \tilde{E}_p(k) \} \circ R_p \quad (14)$$

여기서 $e_p(k) = d(k) - y(k)$,

d : 바람직한 출력,

$$\dot{e}_p(k) = e_p(k) - e_p(k-1) ,$$

$$R_p = E_{pk}(k) \times \tilde{E}_{pk}(k) \times \Delta U(k) \quad \text{이다.}$$

또한 자동 학습 규칙 R_p 는 미리 설정된 바람직한 출력 궤적 d 에 실제 출력이 접근 하도록 FLC에서 사용한 제어 규칙을 출력이 ZE가 되는 영역을 약간 넓게 수정하여 표2와 같이 구성한다.

한편, 오프라인(Off-Line)으로 λ 를 조정할때 현재 출력을 평가할 수 있는 성능 평가 지수 J 가 필요하다. 이 평가 지수 J 는 다음식과 같은 바람직한 출력과 실제 출력과의 절대 오차의 평균값(average of absolute differences between desired output and actual output)을 이용한다.

$$J = \sum_n | d(n) - y(n) | / n \quad (15)$$

위의 식과 같은 성능 평가 지수를 이용한 이유는 과도 상태에서 미리 설정한 기준 출력에 현재의 출력이 얼마나 접근했는가를 파악하기 위해서이다. 즉 J 가 0에 가까운 경우에는 제어 규칙을 수정하지 않아야 하므로 λ 를 작게 해야하고 J 가 클수록 제어 규칙을 많이 수정해야 하므로 λ 를 크게 해야 한다. λ 를 가변시키는 규칙은 다음과 같다.

```
IF J IS PB THEN λ IS PB
IF J IS PS THEN λ IS PS
IF J IS ZE THEN λ IS ZE
```

한편 FLC에서 언어 변수값의 폭이 넓은 경우에는 분해능(Resolution)이 저하되어 정밀 제어가 곤란하다는 문제가 발생한다. 이를 보완하기 위하여 Yoshida는 [15]에서 ZE 부근의 분해능을 높은 언어 변수값을 제안하였고, Li는 [14]에서 오차가 큰 영역과 작은 영역을 분리하여 제어 규칙과 언어 변수값을 이원화하는 방법을 제안하였다. 그러나 LUT의 특성상 제어 입력은 다중 릴레이 시스템(Multi-Relay System)의 출력과 마찬가지로 연속적이다[13]. 따라서 분해능을 높이기 위해 언어 변수값의 폭을 좁게 만들었을 경우 (6)식으로 얻어지는 노이즈가 포함되지 않았을 때의 FLC출력 u 는 다음의 (16)으로 얻을 수 있는 입력 검출시의 노이즈 $\eta(k)$ 가 포함된 경우의 출력 u' 와 큰 차이가 생긴다. 즉 제어 입력이 노이즈 외란등에 민감하게 변화한다.

$$u'(k) = T \{ \psi_{sy} \{ \alpha_2 \cdot [\dot{e}(k) + \eta(k)] \}, \psi_{sy} \{ \alpha_1 \cdot [e(k) + \eta(k)] \} \} \quad (16)$$

따라서 LUT를 이용한 FLC로는 외란과 노이즈가 심한 시스템의 정밀 제어가 어려우므로 본 논문에서는 목표 지점 부근에서는 연속적인 제어 입력을 얻을 수 있는 제어 알고리즘을 사용하고 있다. 즉 과도 상태에서는 표1의 제어 규칙을 갖는 FLC를 정밀 제어가 요구되는 목표 지점 부근에서는 연속적인 제어 입력을 생성하는 PID 제어기를 사용하는 다음과 같은 하이브리드 제어를 이용한다.

$$u(k) = \begin{cases} f[e(k), \dot{e}(k)] , & |e(k)| \geq \epsilon \\ p[e(k), \dot{e}(k)] , & |e(k)| < \epsilon \end{cases} \quad (17)$$

여기서 $f(\cdot)$: 퍼지 제어기,

$p(\cdot)$: PID 제어기,

ϵ : 도립 진자의 경우 선형 특성 구간으로

$\sin\theta = \theta$ 인 구간이다.

3. 자동 학습을 위한 초기 FLC 이득 상수들의 선택 방법

본 절에서는 SOFC를 이용 제어 규칙을 학습하기 이전에 제어 규칙의 빠른 수렴 속도를 위한 FLC의 초기 이득상수들의 결정 방법을 제시하고자 한다.

FLC의 $t = kT$ 에서의 출력은 식(2)의 추론을 거쳐 생성된 LUT를 이용하고, 제어 입력은 FLC의 출력 $u(k)$ 에 출력 이득 상수 α_3 가 곱해서 제어 대상 시스템에 가해진다. 또한, 추론 과정에서 $E(k)$ 와 $E(k)$ 의 소속 함수는 다음과 같다.

$$\mu_E(k)(e) = \begin{cases} 1, & e = \psi_{sy} \{ \alpha_1 \cdot e(k) \} \\ 0, & \text{그 이외의 경우,} \end{cases} \quad (18)$$

$$\mu_{\dot{E}}(k)(\dot{e}) = \begin{cases} 1, & \dot{e} = \psi_{sy} \{ \alpha_2 \cdot \dot{e}(k) \} \\ 0, & \text{그 이외의 경우,} \end{cases}$$

따라서 FLC에는 조정해야할 세개의 이득 상수($\alpha_1, \alpha_2, \alpha_3$)를 포함하고 있다.

[12]에서는 제어 규칙의 상태도에서의 출력 곡선을 분석하여 이득 상수들을 경험적으로 조정하는 방법을 제안하였다. 그러나 실제 컴퓨터를 이용한 실시간 제어에 이용되는 것은 제어 규칙의 상태도가 아니고 LUT이므로 이 LUT위에 출력 곡선을 도시하여 다음의 방법으로 이득 상수를 조정하는 것이 합리적이라 생각된다.

먼저 FLC의 출력 이득 상수 α_3 는 LUT의 최대 제어 입력값 u_{rmax} 가 실제 시스템의 최대 제어 입력값 u_{max} 에 부합할 수 있도록 다음과 같이 조정한다.

$$\alpha_3 = u_{rmax} / u_{max} \quad (19)$$

α_1 과 α_2 는 PD(Proportional Derivative)제어의 P 이득 상수와 D 이득 상수에 해당하는 역할을 하는 것으로 생각할 수 있다. 그리고 LUT는 e 와 \dot{e} 로 구성된 위상 평면(Phase Plane)으로 생각할 수 있다. 그러므로 LUT위에 그림2-(a)와같은 바람직한 위상 곡선(Phase Trajectory)를 도시하고 주어진 α_1 과 α_2 에 대한 실제 위상 곡선과 비교하여 α_1 과 α_2 를 조정하는 방법을 제시한다. 구체적으로 출력 궤적이 그림2-(b)와같이 오차의 전범위 -최대 또는 최소값에서 목표 상태까지- 를 거치지 않았을 경우는 $e(k)$ 가 너무 작게 평가되어 제어입력이 작게 생성된다. 따라서 도달 시간이 길게되므로 $e(k)$ 를 크게 평가하기 위해 α_1 을 증가시켜야 한다. 그림2-(c)와 같이 출력 궤적이 오차의 전범위를 거치지만 오차의 변화량의 전범위를 거치지 않은 이유는 $\dot{e}(k)$ 이 작게 평가되었기 때문이므로 PD 제어의 미분 이득 상수가 작게 선택된 경우와 마찬가지로 진동이 발생한다. 따라서 α_2 를 증가시켜야 한다. 이와는 반대로 출력 곡선이 그림2-(d)와 같을때는 α_1 를 감소시키고, 그림2-(e)와 같을때는 α_2 를 감소시켜야 한다. 만약 목표값 부근에서의 정밀 제어를 위해 [14]와 같이 또다른 형태의 제어 규칙을 추가했을 경우는 이 정밀 제어를 위한 LUT 위에서의 출력 특성이 그림3과 같이 형성되도록 입력 이득 상수들을 조정하여야 한다.

4. 시뮬레이션 및 실험 결과

실험과 시뮬레이션 결과는 다음의 세부류로 구성된다.

첫째 기존의 SOFC와 본 논문에서 제안한 SOFC를 이용했을 때 각각의 학습 횟수에 따른 성능 지수의 변화 상태를 시뮬레이션 결과를 통해 비교한다. 둘째 학습 이전과 학습 이후의 시간에 따른 출력 특성을 시뮬레이션과 실험을 통해 비교 성능이 개선됨을 보인다. 셋째 Li[14]가 제안한 이원화된 제어 규칙을 사용했을 경우와 본 논문에서 제안한 SOFC + PID 제어기의 성능을 실험을 통해 비교한다.

도립 진자의 제어를 위한 전체 시스템은 그림4와 같이 구성되어 있고 그림5는 실제 모습이다. 본 논문에서 사용한 도립 진자의 파라메타는 다음과 같다.

- 수레의 질량 : 2.5 Kg
- 도립 진자의 질량 : 9.5 Kg
- 무게 중심의 높이 : 0.45m
- 마찰 계수 : 시뮬레이션시 이를 무시한다.

실험을 하는데 샘플링 간격은 20msec로, FLC의 초기 이득 상수 α_1 , α_2 및 α_3 는 3절에서 제시한 방법을 통해 각각 2.0, 0.4, 0.6으로 선택하였다.

그림 6은 λ 의 크기에 따른 FLC 성능 지수들의 변화 상태이다. λ 가 0.5일 때는 성능 지수가 점근적인 수렴이 되지 않고 진동하고 λ 가 0.1일 때는 점근적으로 수렴하지만 수렴 속도가 지연되는 결과를 볼 수 있다. 그림7-(a)는 λ 를 표3의 규칙에 의해 변화시켰을 때의 결과이다. 성능 지수가 점근적으로 수렴하면서 수렴 속도가 λ 를 고정시켰을 때보다 빨라졌음을 알 수 있다.

그림7-(b)는 본 논문에서 제안한 SOFC를 이용 도립 진자를 제어한 시뮬레이션 결과이다. 학습 이전의 성능지수는 ITAE = 1.4858, IAE = 9.4078 이고 24회 학습이후 제어 규칙이 수렴하였을 때의 성능지수는 ITAE = 0.5104, IAE = 6.2577이다. 그림8은 초기값이 45°, 30°, 20°에서 학습시킨 LUT를 이용한 실험 결과와 학습 이전의 LUT를 이용한 결과를 비교한 것이다. 도달 시간(Settling Time)이 빨라지면서도 과도 응답(Overshoot)이 발생하지 않음을 볼 수 있다.

그림9는 Li가 [13]에서 제안한 방법으로 도립 진자를 제어한 실험 결과이고 그림10은 본 논문에서 제안한 FLC와 PID 제어를 혼용한 알고리즘을 사용한 실험 결과이다. 실험 결과에서 나타난 것과 같이 FLC를 불안한 시스템의 정밀 제어에 이용하면 정상 상태에서의 오차가 크고 진동함을 알 수 있다. 따라서 그림10과 같이 정상 상태에서 안정한 결과를 얻기 위해서는 연속적인 제어 입력값을 얻을 수 있는 선형 제어 알고리즘을 사용하는 것이 바람직하다. 그림11은 도립 진자가 초기 30°에서 0°로 점근하는 실험 장면이다.

5. 결론

본 논문에서는 퍼지 제어 알고리즘이 불안정한 시스템을 정밀 제어 하는데 부적합하다는 것을 보이고, FLC와 PID제어의 상호 보완적인 측면을 제시 이를 도립 진자의 제어 실험을 통해 입증하였다. 또한 퍼지 제어기의 성능을 좌우하는 이득 상수들에 대한 체계적인 조정 방법을 제시 하였으며 점근적으로 수렴하고 수렴속도를 향상시킨 자동 학습 퍼지 제어기를 구현 하였다.

자동 학습 퍼지 제어기는 스스로 제어 성능을 향상 시킴으로써 엔지니어가 지루한 반복 작업을 통해 제어 규칙을 수정해야 하는 수고를 덜어줄 수 있다. 그러나 SOFC에는 엔지니어가 수행 작업을 통해 조정해야 하는 파라메타들이 많이 있다. 따라서 이들 파라메타들과 출력 특성간의 분석을 통해 이들을 자동 조정

할 수 있는 알고리즘에 대한 연구가 필요하다. 또한 SOFC의 안정도에 대한 수학적 분석이 필요하다.

Reference

- [1] L.A. Zadeh, "Fussy Sets," Inform. Contr., vol.8, pp.338-353, 1965.
- [2] H.J. Zimmermann, "Decision Making in Fuzzy Environment", Kluwer-Nijhoff Publishing in Boston, pp.213-260, 1986.
- [3] L.A. Zadeh, "Making Computers Think Like People," IEEE Spectrum, August 1984.
- [4] _____, "Outline of a New Approach to the Analysis of Complex Systems and Decision Process," IEEE Trans. Syst., Man, Cybern., vol.3, no.1, pp.28-44 January 1973.
- [5] M. Sugeno, "Fussy Control of Model Car," Fussy Set and System, vol.16, pp.103-113, 1985.
- [6] C.C. Lee, "Fuzzy Logic in Control Systems: Fuzzy Logic Controller-Part I and II," IEEE Trans. Syst., Man, Cybern., vol.20, no.2, pp.404-435, March/April 1973.
- [7] T.J. Procyk, E.H. Mamdani, "A Linguistic Self-Organizing Process Controller," Autometica, vol.15, pp.15-30, 1979.
- [8] T. Yamazaki, M. Sugeno, "Self-Organizing Fuzzy Controller," Trans. SICE(in Japanese), vol.20, no.8, pp.720-726, 1984.
- [9] S. Shao, "Fuzzy Self-Organizing Controller and It's Application for Dynamic Process," Fuzzy Set and Systems, vol.26, pp.151-164, 1988.
- [10] M. Maeda, S. Murakami, "Self Tuning Fuzzy Logic Controller," Trans. SICE(in Japanese), vol.24, no.2, pp.191-197, 1988.
- [11] R. Tansheit, E.M. Sharf, "Experiments With The Use of A Rule-Based Self-Organizing Controller For Robotics Applications," Fuzzy Set and Systems, vol.26, pp.195-214, 1988.
- [12] M. Braae, D.A. Rutherford, "Selection of Parameters for a Fuzzy Logic Controller," Fuzzy Set and Systems, vol.2, pp.185-199, 1979.
- [13] _____, "Theoretical and Linguistic Aspects of the Fuzzy Logic Controller," Autometica, vol.15, pp.553-577, 1979.
- [14] Y.F. Li, C.C. Lau, "Development of Fuzzy Algorithms for Servo Systems," IEEE Contr. Syst. Magazine, vol.9, pp.65-72, 1989.
- [15] M. Yoshida, Y. Tsutsumi, "Gain Tuning Method For Design of Fuzzy Control Systems," Proc. Int. Conf. on Fuzzy Logic & Neural Networks, pp.405-408, 1990.

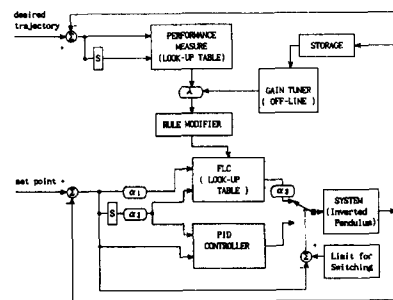
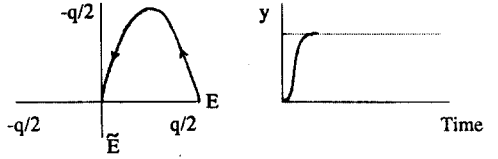
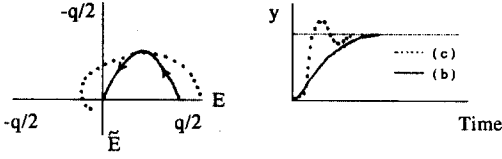


그림1. 수렴성의 개선을 위한 SOFC의 구조
Fig.1. Structure of SOFC to Improve Convergency

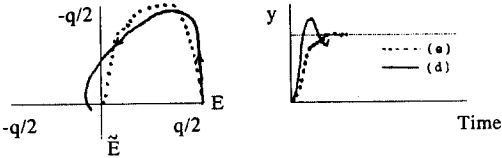


(a) 바람직한 출력 궤적(Desired Output Trajectory)



(b) α_1 이 작다

(c) α_2 가 작다



(d) α_1 이 크다

(e) α_2 가 크다

그림2. LUT에서의 출력 궤적과 시간 응답.
Fig.2. Output Trajectory on LUT and Time Response.

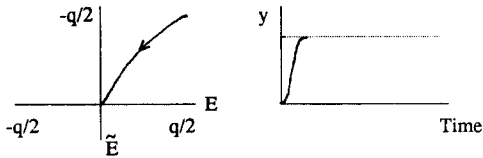


그림3. 미세 조정을 위한 LUT에서의 출력 궤적과 시간 응답.
Fig.3. Output Trajectory on LUT to Fine Control and Time Response.

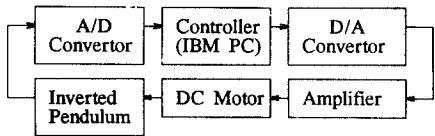


그림4. 도립 진자 제어 시스템
Fig.4. Inverted Pendulum Control System

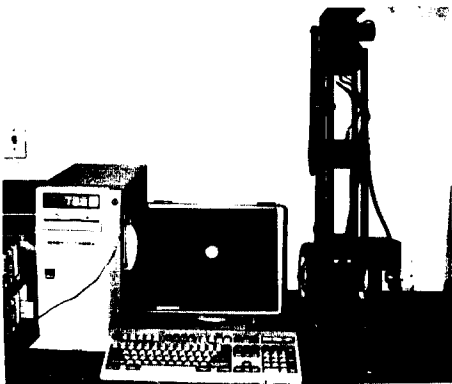


그림5. 시스템의 실제 모습
Fig.5. Real Figure of System

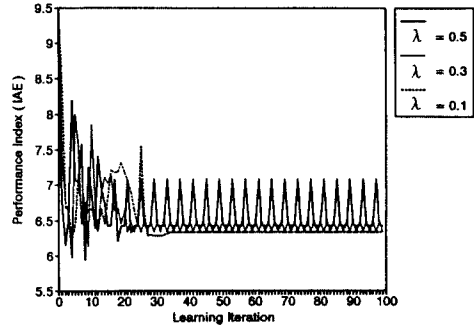
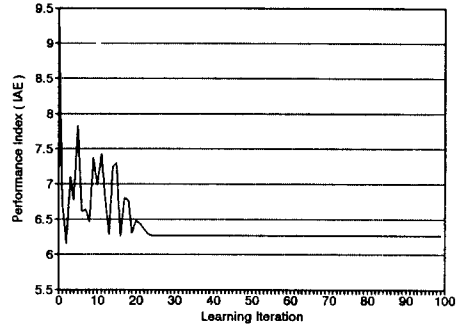
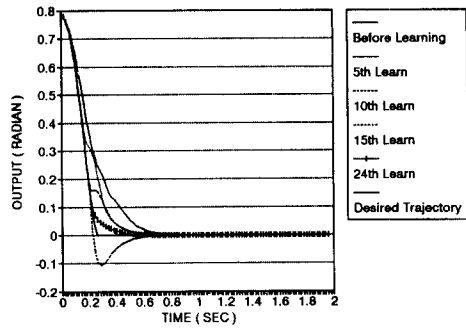


그림6. λ 의 변화에 따른 성능 지수 (IAE)
Fig.6. Performance Index Result of Variation of λ



(a) 성능 지수(IAE)



(b) 시간 응답(Time Response)

그림7. SOFC의 성능 지수와 시간 응답
Fig.7. Performance Index and Time Response of SOFC

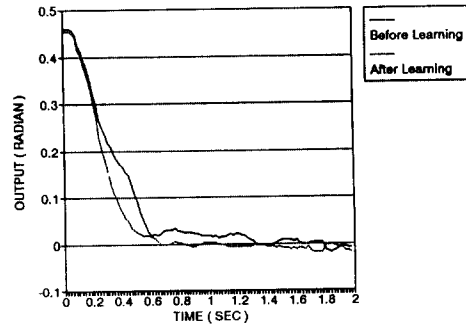
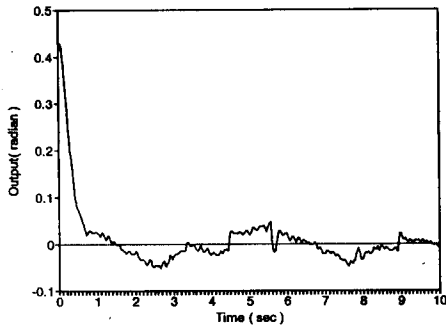
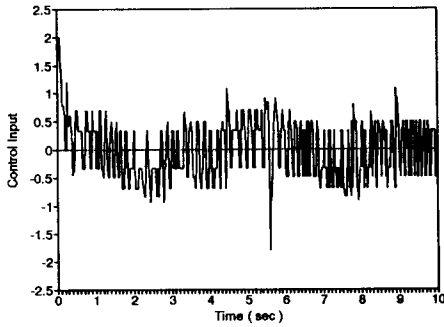


그림8. SOFC를 이용한 실험 결과
Fig.8. Experiments Result of SOFC

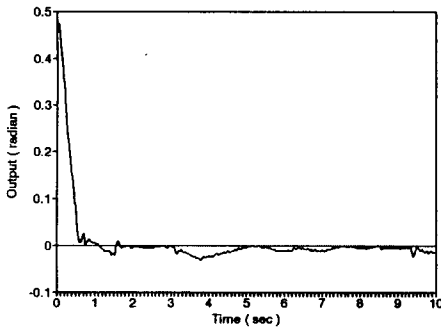


(a) 시간 응답 (Time Response)

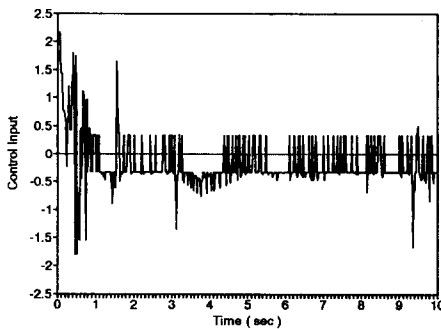


(b) 제어 입력 (Control Input)

그림9. 이원화된 제어규칙을 이용한 실험 결과
Fig.9. Experiments Result using Dual Control Rules

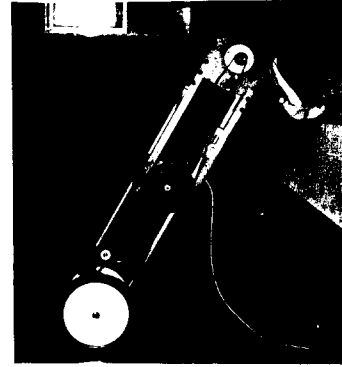


(a) 시간 응답 (Time Response)

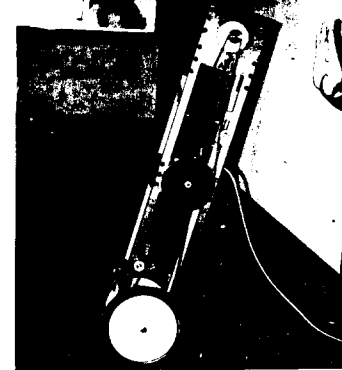


(b) 제어 입력 (Control Input)

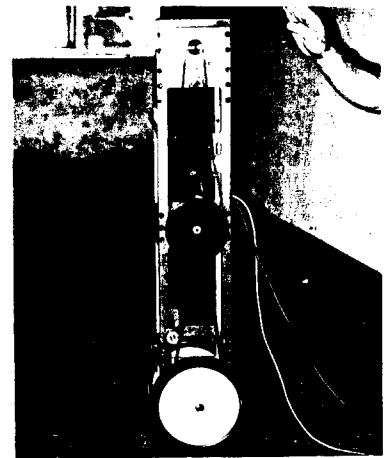
그림10. 퍼지 + PID 하이브리드 제어기를 이용한 실험 결과
Fig.10 Experiments Result using SOFC Plus PID Controller



(a) 초기 상태 (Initial State)



(b) 과도 상태 (Transient State)



(c) 정상 상태 (Steady State)

그림11. 도립 진자가 직립하는 모습
Fig.11. Erection of Inverted Pendulum

표1. FLC에 이용한 제어 규칙 Table1. Control Rules in FLC
표2. 자동학습 규칙 Table2. Self-Organizing Rules

		E							
		NB	NM	NS	ZE	PS	PM	PB	
E	NB	NB	NB	NM	NS	ZE	PS	PM	PB
	NM	NB	NB	NM	NS	ZE	PS	PM	PB
	NS	NB	NM	NM	NS	ZE	PS	PM	PB
	ZE	NB	NM	NM	NS	ZE	PS	PM	PB
	PS	NM	NS	ZE	PS	PM	PB		
PM	NS	ZE	PS	PM	PB				
PB	ZE	PS	PM	PB					

		E _p						
		NB	NM	NS	ZE	PS	PM	PB
E _p	NB	NB	NB	NM	NS	ZE	PS	PB
	NM	NB	NB	NM	NS	ZE	PS	PB
	NS	NB	NM	NM	NS	ZE	PS	PM
	ZE	NB	NM	NS	ZE	PS	PM	PB
	PS	NM	NS	ZE	PS	PM	PB	
PM	NS	ZE	PS	PM	PB			
PB	ZE	PS	PM	PB				