

룩업 테이블을 이용한 자동 학습 퍼지 제어기의 설계에 관한 연구

(Design of a Self-Organizing Fuzzy Controller Using the Look-Up Tables)

李 鏞 魯*, 金 兌 垣*, 徐 一 弘*

(Yong No Lee, Tae Won Kim, and Il Hong Suh)

要 約

본 논문에서는 룩업 테이블을 이용한 자동 학습 퍼지 제어기와 비례 미분 제어기가 혼합된 하이브리드 제어기를 제안한다. 제안된 자동 학습 퍼지 제어기는 전형적인 퍼지 추론 방법을 이용한 퍼지 제어기와 자동 학습부로 구성되며 자동 학습부는 다시 온라인(On-Line)으로 룩업 테이블의 출력을 자동 학습을 통해 수정하는 알고리즘과 제어 성능을 평가한 후 수렴성의 개선을 위해 룩업 테이블 출력의 수정 정도를 결정하는 알고리즘으로 구성된다. 그리고 목표 지점에서의 정밀 제어를 위해서 정밀 제어를 요하는 구간에서는 자동 학습 퍼지 제어기를 비례 미분 제어기로 대체한다. 여기서 비례 미분 제어기의 이득 상수값은 시스템의 출력이 미리 지정해 놓은 구간으로 진입할 때의 오차의 속도를 조건부로 하는 퍼지 규칙에 의해 결정된다. 제안된 제어기의 효율성을 보이기 위해 도립 진자를 대상으로 한 시뮬레이션과 실험을 통해 성능을 분석한다.

Abstract

A novel self-organizing fuzzy plus PD control algorithm is proposed, where the proposed 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 Look-Up Table(LUT) for the fuzzy control 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 PD controller in a prespecified region nearby the set point for good settling actions, where PD gain parameters are determined by fuzzy rules based on the magnitude of error velocity at the instant when the output penetrates into the prespecified region. To show the effectiveness of the proposed controller, extensive computer simulation results as well as experimental results are illustrated for an inverted pendulum system.

I. 서 론

*正會員, 漢陽大學校 電子工學科
(Dept. of Elec. Eng., Hanyang Univ.)
接受日字: 1991年 12月 3日

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 활발하게 연구되는 분야가 되었고, FLC의 일반적인 구조와 제어 규칙의 구성 방법, 추론 방법들이 제시되었다.^[4-6] FLC가 제어 엔지니어들의 관심을 끄는 이유는 FLC 알고리즘이 경험이나 직관등에 의한 주먹 구구식의 규칙을 언어 변수로 표현하고 구현할 수 있으며 제어 대상 시스템의 수학적 모델링(Modelling)이 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 시행 착오적으로 수정 해야한다는 단점을 가지고 있다. Mamdani 와 Procyk^[7]는 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(Sampling) 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 향상시킬수 있는 자동 학습 퍼지 제어기(SOFC)를 제안하고 이를 그림 1과 같은 도립 진자를 대상으로한 시뮬레이션과 실험을 통해 성능이 개선됨을 보인다. 그러나 초기 제어 규칙에 따라 대부분의 SOFC가 그렇듯이 학습이 되지않을 수가 있다. 따라서 자동학습 이전 상태의 FLC 이득 상수들을 경험적 방법에 의해 조심스럽게 결정하여야 한다. Braae 및 Rutherford는 [12][13]에서 제어 규칙의 상태도(State Plane)위에 제어 대상 시스템의 출력 궤적(Trajectory)을 도시하고 이를 분석하여 바람직한 출력 궤적을 형성하도록 제어 규칙과 제어기의 이득 상



그림 1. 도립 진자 제어 시스템

Fig. 1. Inverted pendulum control system.

수를 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 제시하고자 한다.

한편 FLC는 노이즈나 제어 대상 시스템의 파라미터의 불확실성 또는 변화에 둔감하다는 장점이 있으나 목표 지점 근처에서의 정밀 제어가 어렵다는 단점이 있다. 이를 보상하기 위하여 Li와 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 접근 할수록 폭이 점점 좁아지는 형태의 언어 변수값을 제안하였다. 그러나 이 두 경우 모두 언어 변수의 폭을 ZE 부근에서 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 PD 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, \tilde{E}, U)은 첨자를 붙이며 어떤 시간에서의 언어 변수의 값 ($E(k), \tilde{E}(k), U(k)$)은 괄호 안에 시간을 표시한다. 또한 \bar{A} 는 A 의 여집합을 의미한다.

II. 자동 학습 퍼지 제어기의 설계

FLC는 제어 규칙을 반복적인 실험을 통해 만족할 만한 성능이 나올때 까지 시행 착오적인 시도 및 수정 과정을 통해 설계를 해야한다. 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 그림2에서 보는바와 같이 일반적 FLC와 비례 미분 제어기로 구성되는 하위층과 제어 규칙의 조정을 담당하는 상위층으로 구성된다.

1. 퍼지 로직 제어기(FLC)

본 논문에서 사용하고자 하는 하위층 FLC의 제어 규칙은 입력 퍼지 변수로써 오차 E 및 오차의 시간에 따른 변화량 \dot{E} 를 이용하고 출력 퍼지 변수로써 U 를 선택할때 입력과 출력의 관계(Relation)가 IF THEN ELSE (IF E is \tilde{E}_i and \dot{E} is \tilde{E}_i THEN U is U_i ELSE...)의 형태로 표 1과 같이 구성된 것이며, 이

표 1. 제어 규칙 R

Table.1. Control rules in FLC R.

		E						
		NB	NM	NS	ZE	PS	PM	PB
\tilde{E}	NB	NB	NB	NM	NM	NS	ZE	ZE
	NM	NB	NM	NM	NS	NS	ZE	PS
	NS	NB	NM	NS	NS	ZE	PS	PM
	ZE	NB	NB	NM	ZE	PM	PB	PB
	PS	NM	NS	ZE	PS	PS	PM	PB
	PM	NS	ZE	PS	PS	PM	PM	PB
	PB	ZE	ZE	PS	PM	PM	PB	PB

러한 제어 규칙을 생성하기 위해서는 먼저 시스템의 동적 특성을 분석하여야 한다. 동적 특성의 분석을 기초로하여, 어떠한 입력에 대해서도 제어가 이루어져야 한다는 제어 규칙의 완전성과 제어 규칙들 사이에 공유하는 영역이 있어야 한다는 상호 연관성, 제어 규칙들 사이에 상호 모순되는 부분이 있어서는 안된다는 일관성등을 고려 하여야 한다. 또한 제어 규칙은 출력이 목표 상태에 도달할수 있도록 결정되어야 하며 이에 대한 방법론은 [5][6]등 많은 연구가 되어 있다. 또한 제어 규칙에서 i번째 규칙의 입출력 관계 R_i 는 다음과 같이 카테시언 프로덕트(Cartesian Product) [7]로 표현된다.

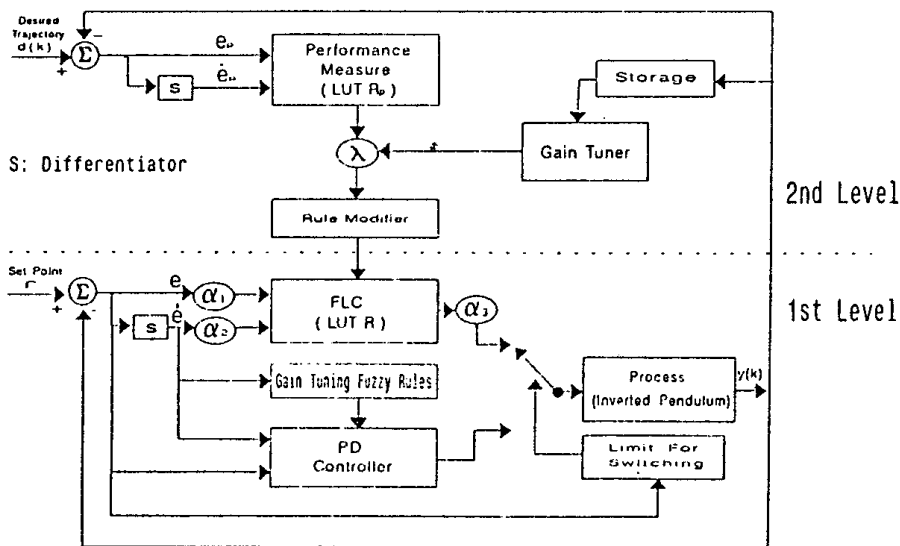


그림 2. SOFC의 블록 다이어그램
Fig. 2. Block diagram of SOFC.

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

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

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

으로 구하며 여기서 \circ 는 Zadeh^[11]의 합성 규칙 (Composition Rule of Inference)인 Sup-Min연산을 이용한다.

한편, 실제 시스템에서의 제어 입력은 퍼지량이 아닌 크리슈 값(Crisp Value)이므로 비 퍼지화(Defuzzification) 과정이 필요하다. 비 퍼지화의 방법으로는 면적 중심법^[6]을 이용하며 $\mu_{u(k)}$ 를 퍼지 집합 $U(k)$ 의 소속 함수로 정의하면 면적 중심법은 다음식과 같이 표현 된다.

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

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

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

이며, 여기서 $\text{Step} = (p_{\max} - p_{\min}) / (q-1)$, $\phi(\cdot)$ 는 절단(Truncation)함수이다. 특히 q 가 짝수이면 0을 중심으로 좌우 대칭이 되도록 정량화 시키는 함수 $\phi_{sv}(p)$ 는

$$\phi_{sv}(p) = \phi(p) - q/2 \quad (5)$$

이다. 정량화 과정에서 p_{\min} 과 p_{\max} 의 값은 시스템의 특성을 미리 파악하여 결정하여야 하며 (4)식에서 p_{\min} 보다 작거나 p_{\max} 보다 큰 p 는 각각 p_{\min}, p_{\max} 와 같은 정량화 단계 0, q 로 정량화 된다. 또한 정량화 과정에 있어서 $q+1$ 개의 각 단계에는 정수로 구성된 이름이 할당되고 이것은 물리량을 이산화한 값이 된다. 이것을 이용하여 언어 변수의 소속 함수는 표3과 같이 이산화한 물리량의 각 단계에 대한 소속 정도로 새롭게 구성된다. 이산화된 물리량에 대한 LUT는 입력 신호들의 발생할수 있는 모든 조합에 대한 제어기의 출력 집합이라고 정의 할 수 있고 이것은 제어기의 수행

시간을 절약하기 위해 오프 라인(Off-Line)으로 계산한다. 여기서 정량화 단계는 연속적인 물리량을 적당히 근사화 할 수 있도록 충분히 커야 하지만 메모리의 절약을 위해서는 작을수록 좋다. 따라서 정량화 단계 $q+1$ 의 값은 어느 정도의 정밀도로 제어를 수행할 것인가에 따라 결정되어야 한다. 이제 (5)식으로 정량화된 입력 변수를 이용하여 (2)식의 추론 과정과 (3)식의 비퍼지화 과정을 거쳐 LUT가 만들어진다.^[11] 이때 LUT를 만들기 위한 추론은 $(q+1) \times (q+1) \times r$, 여기서 r 은 제어 규칙의 수, 만큼 수행되며 LUT를 저장하기 위한 기억장소는 $(q+1) \times (q+1)$ 개가 필요하다. 결국 $t=kT$ 에서 FLC의 출력 $u(k)$ 는

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

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

2. 자동 학습 알고리즘

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.

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

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

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

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

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 오차 $e(k)$ 와 시간에 따른 오차의 변화량 $\dot{e}(k)$ 이 발생한 이유는 바로 이전의 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 수정하는 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 규

칙을 수정할 경우 매 샘플링 시간마다 추론을 다시 해야 하므로 실시간 제어에 어려움이 따른다. 따라서 실시간 제어를 위해서는 제어 규칙을 수정하는 것보다 LUT의 출력값을 수정하는 방법이 합리적이며 동시에 출력값의 수정량도 가변시키는 방법을 이용해야 한다.

본 논문에서는 그림1의 상위층을 온라인(On-Line)으로 록업 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 록업 테이블은 다음과 같은 수정 방법을 이용 온라인(On-Line)으로 수정한다.

$$T'(\dot{e}, e; k) = T(k)(\dot{e}, e; k) + \Delta u(k) \cdot \lambda \quad (11)$$

$$T(; k+1) = \{ T(; k) \cap \bar{T} \{ \phi_{y1}[\alpha_2 \cdot \dot{e}(k-1)], \phi_{y1}[\alpha_1 \cdot e(k-1)k]; \} \cup T' \{ \phi_{y2}[\alpha_2 \cdot \dot{e}(k-1)], \phi_{y2}[\alpha_1 \cdot e(k-1)]; k \} \} \quad (12)$$

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 수정 해야 할 양의 퍼지 집합을 ΔU 라 표시하면 k 번째 샘플링 시간에서의 실 수정량 $\Delta u(k)$ 는 다음 식과 같이 얻을수 있다.

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

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

여기서 $e_p(k) = d(k) - y(k)$,
 d : 바람직한 출력,
 $\dot{e}_p(k) = e_p(k) - e_p(k-1)$,
 $R_p = E_{p1} \times \tilde{E}_{p1} \times \Delta U_1$ 이다.

자동 학습 규칙 R_p 는 제어 규칙 R 을 이용한 시스템의 실제 출력이 미리 설정된 바람직한 출력 d 에 근접하도록 하기 위한 것으로 ZE 영역이 약간 넓다는 것을 제외하고는 표1의 제어 규칙과 거의 유사하다. [8-11]에서는 자동 학습 규칙을 생성하기 위해 입력 변수로써 E 와 \tilde{E} 를 이용하였다. 그리고 자동학습 규칙의 이름을 'Conditional Statements performance Index', [8] 'The Performance Index Measure Rule'[9]라고 각각 명명 하였다. 이들의 경우 학습의 개념이 출력을 목

표 값에 빨리 도달 시키게 하는 것이므로 오차가 작은 영역에서는 NB로 오차가 큰 영역에서는 PB로 제어 규칙의 출력이 학습된다. 따라서 제어 입력이 시간 최적 제어(Time Optimal Control)의 제어 입력과 유사하게 되어 리미트 사이클(Limit Cycle) 현상과 같이 바람직하지 못한 현상이 발생한다. 따라서 입력 변수로는 실제 출력과 바람직한 출력간의 오차 E_p 와 오차의 변화량 \tilde{E}_p 를 선택했으며, 학습 규칙 자체는 생성 방법에 있어서 제어 규칙의 생성 방법과 차이가 없다. 여기서 자동 학습 규칙의 출력이라는 ZE라는 것은 사용된 제어 규칙을 수정할 필요가 없다는 것을 뜻하고 PB라는 것은 제어 규칙의 출력을 더 크게 해야 한다는 것을 뜻한다. 그러나 자동 학습 규칙이 E_p 와 \tilde{E}_p 로 구성되어 있기 때문에 시 변화(Time Varying) 목표 입력에 대해서는 수렴성이 보장되지 못한다.

표 2. 자동 학습규칙 R_p
 Table.2. Self-organizing rules R_p .

		E_p						
		NB	NM	NS	ZE	PS	PM	PB
\tilde{E}_p	NB	NB	NB	NB	NM	ZE	ZE	ZE
	NM	NB	NB	NM	NM	ZE	ZE	ZE
	NS	NB	NM	NS	ZE	ZE	PS	PM
	ZE	NB	NM	NS	ZE	PS	PM	PB
	PS	NM	NS	ZE	ZE	PS	PM	PB
	PM	ZE	ZE	ZE	PM	PM	PB	PB
	PB	ZE	ZE	ZE	PM	PB	PB	PB

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 수정

해야 하므로 λ 를 크게 해야 한다. λ 를 가변시키는 규칙은 다음과 같다.

$$\begin{aligned} & \text{IF } J \text{ IS } PB \text{ THEN } \lambda \text{ IS } PB \\ & \text{IF } J \text{ IS } PS \text{ THEN } \lambda \text{ IS } PS \\ & \text{IF } J \text{ IS } ZE \text{ THEN } \lambda \text{ IS } ZE \end{aligned} \quad (16)$$

3. 정밀 제어를 위한 비례 미분(PD) 제어기

한편 FLC에서 언어 변수값의 폭이 넓을 경우에는 분해능(Resolution)이 저하되어 정밀 제어가 곤란하다는 문제가 발생한다. 이를 보완하기 위하여 Yoshida는 [15]에서 그림3(a)와 같이 ZE 부근의 분해능을 높인 언어 변수값을 제안하였고, Li는 [14]에서 그림3(b)와 같이 오차가 큰 영역과 작은 영역을 분리하여 제어 규칙과 언어 변수값을 이원화하는 방법을 제안하였다.

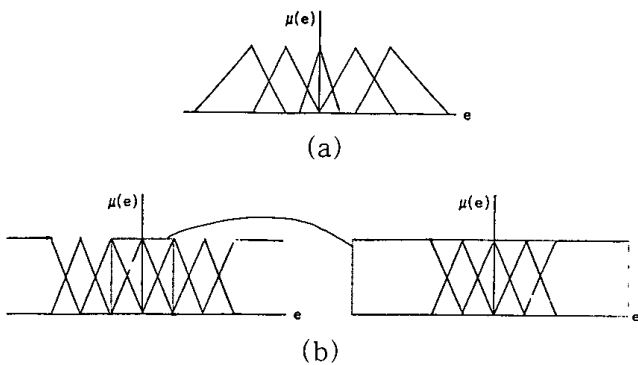


그림 3. 분해능을 높이기 위해 제안된 언어 변수값
(a) [15]에서 제안한 언어 변수값의 모양
(b) [14]에서 제안한 언어 변수값의 모양
Fig. 3. Linguistic values proposed to obtain high resolution.
(a) Linguistic values proposed in [15],
(b) Linguistic values proposed in [14].

그러나 LUT의 특성상 제어 입력은 그림4-(a)과 같은 다중 릴레이 시스템(Multi-Relay System)의 출력과 마찬가지로 단속적이다.^[13] 따라서 분해능을 높이기 위해 언어 변수값의 폭을 그림4(b)와 같이 좁게 만들었을 경우 (6)식으로 얻어지는 노이즈가 포함되지 않았을 때의 FLC출력 u 는 다음 (16)으로 얻을수 있는 입력 검출시의 노이즈 $\eta(k)$ 가 포함된 경우의 u' 와 큰 차이가 생긴다. 즉 제어 입력이 노이즈 외란등에 민감하게 변화한다.

$$u'(k) = T \left\{ \phi_{zy} \{ \alpha_2 \cdot [\dot{e}(k) + \eta(k)] \}, \phi_{zy} \{ \alpha_1 \cdot [e(k) + \eta(k)] \} \right\} \quad (17)$$

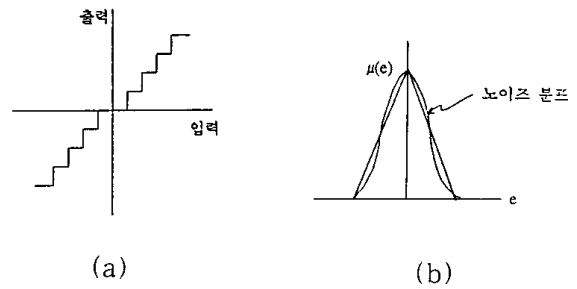


그림 4. FLC를 이용한 정밀 제어를 어렵게 만드는 요소들
(a) 다중 릴레이 시스템의 출력
(b) 노이즈에 민감한 언어 변수값
Fig. 4. Elements prevents FLC from fine-controlling of the system.
(a) Output of the Multi-relay system,
(b) Linguistic values sensitive to noise.

따라서 LUT를 이용한 FLC로는 외란과 노이즈가 심한 시스템의 정밀 제어가 어려우므로 본 논문에서는 목표 지점 부근에서는 연속적인 제어 입력을 얻을수 있는 제어 알고리즘을 사용하고자 한다. 즉 과도 상태에서는 표1의 제어 규칙을 갖는 FLC를 정밀 제어가 요구되는 목표 지점 부근에서는 연속적인 제어 입력을 생성하는 PD제어기를 사용하는 다음과 같은 하이브리드 제어(Hybrid Control)를 이용한다.

$$u(k) = \begin{cases} f[e(k), \dot{e}(k)], & |e(k)| \geq \epsilon \\ p[e(k), \dot{e}(k)], & |e(k)| < \epsilon \end{cases} \quad (18)$$

여기서 $f(\cdot)$: 퍼지 제어기
 $p(\cdot)$: PD 제어기,
 ϵ : 정밀 제어가 요구되는 구간이다.

그러나 이와 같은 하이브리드 형태의 제어 알고리즘을 이용할 경우 SOFC 알고리즘에서 PD 알고리즘으로 전환하는 경계 부근에서 제어 대상의 상태 \dot{e} 에 따라 정밀 제어의 성능이 영향을 받게 된다. 이러한 문제를 극복하기 위해 본 논문에서는 \dot{e} 에 따라서 동작점 세개를 선택하고 각각에 대해 PD 제어기의 이득 상수들을 조정후 다음과 같은 퍼지 규칙을 이용하여 제어 입력을 결정한다. 여기서 결과부의 u_i 는 퍼지 싱글톤(Singleton)이다.

$$\begin{aligned} & \text{IF } |\dot{e}| \text{ IS } PB \text{ THEN } u_1 = K_{p1} \cdot e + K_{d1} \cdot \dot{e} \\ & \text{IF } |\dot{e}| \text{ IS } PS \text{ THEN } u_2 = K_{p2} \cdot e + K_{d2} \cdot \dot{e} \\ & \text{IF } |\dot{e}| \text{ IS } ZE \text{ THEN } u_3 = K_{p3} \cdot e + K_{d3} \cdot \dot{e} \end{aligned} \quad (19)$$

즉, 세개의 동작점에서 제어 곡선(Control Curve)

들이 퍼지 추론을 통해 보간(Interpolation)되어 제어 평면(Control Surface)을 형성하므로 |E|에 따라 비선형적인 PD제어기를 형성하게 된다.

Ⅲ. 자동 학습을 위한 초기 FLC이득 상수들의 선택 방법

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

$t=kT$ 에서 FLC의 출력은 식(2)의 추론을 거쳐 생성된 LUT를 이용하고, 제어 입력은 FLC의 출력 $u(k)$ 에 출력 이득 상수 α_3 를 곱해서 제어 대상 시스템에 가해진다. 또한 추론 과정에서 $E(k)$ 와 $\tilde{E}(k)$ 의 소속함수는 다음과 같다.

$$\mu_{Ekk(e)} = \begin{cases} 1, & e = \phi_{sy}\{\alpha_1 e(k)\} \\ 0, & \text{그 이외의 경우,} \end{cases} \quad (20)$$

$$\mu_{\tilde{E}kk(e)} = \begin{cases} 1, & \dot{e} = \phi_{sy}\{\alpha_1 \cdot \dot{e}(k)\} \\ 0, & \text{그 이외의 경우,} \end{cases}$$

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

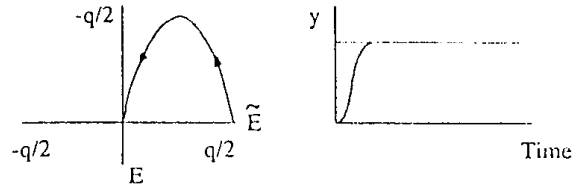
[12]에서는 제어 규칙의 상태도에서의 출력 곡선을 분석하여 이득 상수들을 경험적으로 조정하는 방법을 제안하였다. 그러나 실제 컴퓨터를 이용한 실시간 제어에 이용되는 것은 제어 규칙의 상태도가 아니고 LUT이므로 이 LUT위에 출력 곡선을 도시하여 다음의 방법으로 이득 상수를 조정하는 것이 합리적이라 생각된다.

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

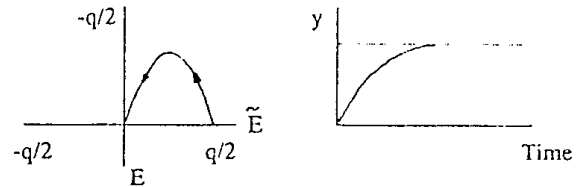
$$\alpha_3 = u_{rmax}/u_{max} \quad (21)$$

α_1 과 α_2 는 PD(Propotional 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 그림5(a)와 같은 바람직한 위상곡선(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. 이때 바람 직한 위상 곡선은 시간 최적 제어시의

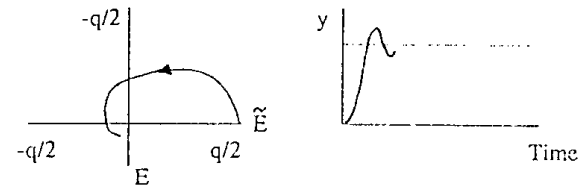
위상 곡선으로 선택한다. 구체적으로 출력 궤적이 그림5(b)와 같이 오차의 전 범위-최대 또는 최소값에서 목표 상태까지를 거치지 않았을 경우는 $e(k)$ 가 너무 작게 평가되어 제어 입력이 작게 생성된다. 따라서 도달시간이 길게 되므로 $e(k)$ 를 크게 평가하기 위해 α_1 을 증가시켜야 한다. 그림5-(c)와 같이 출력 궤적이 오차의 전범위를 거치지만 오차의 변화량의 전범위를 거



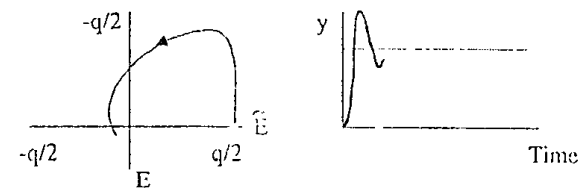
(a) 바람직한 출력 궤적(Desired Output Trajectory)



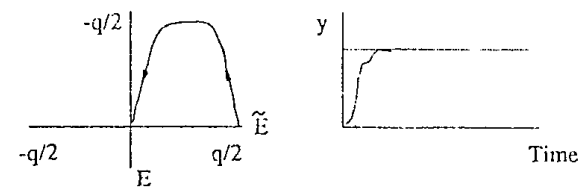
(b) α_1 이 작다(α_1 is too small.)



(c) α_2 가 작다(α_2 is too small.)



(d) α_1 이 크다(α_1 is too large.)



(e) α_2 이 크다(α_2 is too large.)

그림 5. 룩업 테이블 위에서의 출력 궤적과 시간 응답

Fig. 5. Output trajectory on LUT and time response.

치지 않은 이유는 $e(k)$ 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 그림5(d)와 같을 때는 α_1 를 감소시키고, 그림5(e)와 같을 때는 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 그림 6과 같이 형성되도록 입력 이득 상수들을 조정하여야 한다.

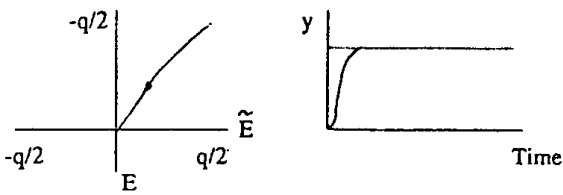


그림 6. 정밀 제어를 위한 룩업 테이블 위에서의 출력 궤적과 시간 응답

Fig. 6. Output trajectory on LUT to fine control and time response.

IV. 시뮬레이션 및 실험 결과

실험 장비는 그림 7에서 보는바와 같이 도립 진자, A/D, D/A 변환기, 서보 모터, IBM PC 386으로 구성된다. 도립 진자에는 포텐시오테타(Potentiometer)와 DC 서보 모터가 장착되어 있으며 진자와 바닥과의 각도는 A/D 변환기를 거쳐 IBM PC 386에 의해 측정된다. 그리고 제어 입력은 D/A 변환기와 증폭기를 거쳐 생성된다. 제안된 알고리즘의 작용을 분석하기 위하여 도립 진자의 운동 상태를 모의 실험한다. 모의 실험에 사용한 도립 진자의 운동 방정식은 다음과 같다.

$$(M + m)\ddot{x} + ml\ddot{\theta}\cos\theta - ml\dot{\theta}^2\sin\theta = u \quad (22)$$

$$4/3ml^2\ddot{\theta} + m\dot{x}\cos\theta - mg\sin\theta = 0$$

여기서 M, m, l, g는 각각 수레의 무게, 진자의 무게, 진자의 높이, 중력이며 θ, x, u 는 각각 도립 진자와 바닥과의 각도, 수레의 위치, 제어입력을 나타낸다. 모의 실험에서 사용한 M, m, l, g의 값은 $M=2.5\text{kg}$, $m=9.5\text{kg}$, $l=0.45\text{m}$, $g=9.8\text{m/sec}^2$ 이다. 제안된 자동 학습 제어기의 이득 상수 $\alpha_1, \alpha_2, \alpha_3$ 는 3절에서 제안한 방법에 의해 각각 1.8, 0.23, 80으로 결정되었고 (19)식의 이득 상수 $K_{pi}, K_{di}(i=1, 2, 3)$ 는 $K_{p1}=1200$, $K_{d1}=71$, $K_{p2}=980$, $K_{d2}=64$, $K_{p3}=650$, $K_{d3}=68$ 으로

결정되었다. 여기서 $K_{pi}, K_{di}(i=1, 2, 3)$ 는 시스템의 오차의 속도가 각각 0, -2, and -4 rad/sec 일때 조정 한 값이다. 정량화 단계와 언어 변수값은 표3, 제어규칙을 이용해 만든 LUT는 표4, 자동 학습 규칙을 이용해 만든 LUT는 표5와 같이 생성된다. 한편 샘플링 시간은 IBM PC 386의 계산 성능과 도립 진자의 특성을 고려하여 20msec로 결정 하였다.

모의 실험과 실험 결과는 다음과 같이 세부류로 구성된다. 첫째, LUT의 수정량을 고정시켰을 경우와 가변시켰을 경우의 SOFC의 성능지수 IAE를 비교한다. 둘째 자동 학습의 효율성을 보이기 위해 SOFC의 초기 수행시의 성능과 자동학습이 완료 되었을 때의 성능을 비교한다. 셋째 [14]에서 제안한 이원화된 제어 규칙을 이용했을 경우의 성능과 본 논문에서 제안된 SOFC의 성능을 비교한다.

그림8과 그림9에서는 λ 를 고정 시켰을 때와 (16)식

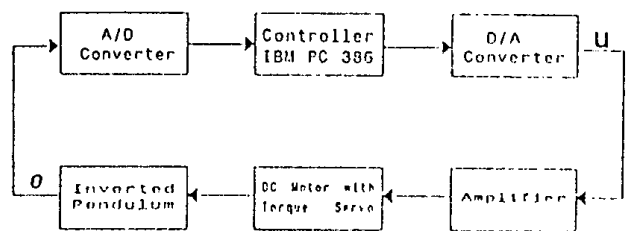


그림 7. 도립 진자 제어를 위한 실험 장치의 구성도

Fig. 7. Experimental workcell for the control of an inverted pendulum system.

표 3. 정량화 단계와 언어 변수값

Table.3. Quantization levels and primary fuzzy set.

Leverl No.	NB	NM	NS	ZE	PS	PM	PB
-6	1.0	0.3	0.0	0.0	0.0	0.0	0.0
-5	0.7	0.7	0.0	0.0	0.0	0.0	0.0
-4	0.3	0.1	0.3	0.0	0.0	0.0	0.0
-3	0.0	0.7	0.7	0.0	0.0	0.0	0.0
-2	0.0	0.3	1.0	0.3	0.0	0.0	0.0
-1	0.0	0.0	0.7	0.7	0.0	0.0	0.0
0	0.0	0.0	0.3	1.0	0.3	0.0	0.0
1	0.0	0.0	0.0	0.7	0.7	0.0	0.0
2	0.0	0.0	0.0	0.3	1.0	0.3	0.0
3	0.7	0.7	0.0	0.0	0.0	0.0	0.0
4	0.0	0.0	0.0	0.0	0.3	1.0	0.3
5	0.0	0.0	0.0	0.0	0.0	0.7	0.7
6	0.0	0.0	0.0	0.0	0.0	0.3	1.0

표 4. 제어 규칙 R에 해당하는 룩업 테이블
Table.4. Look-up table for control rules R.

\dot{e} e	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6
-6	-5	-5	-5	-4	-3	-3	-3	-3	-2	-1	0	1	1
-5	-5	-4	-4	-4	-3	-3	-3	-3	-2	-1	0	1	1
-4	-5	-4	-3	-3	-3	-3	-2	-2	-1	0	1	2	2
-3	-5	-4	-3	-3	-3	-3	-2	-1	0	0	1	2	2
-2	-5	-4	-3	-3	-2	-2	-1	0	1	1	1	2	3
-1	-4	-4	-3	-3	-2	-2	0	1	1	2	3	3	4
0	-4	-4	-3	-3	-2	-1	0	1	2	3	3	4	4
1	-4	-3	-3	-2	-1	-1	0	2	2	3	4	4	5
2	-3	-2	-1	-1	-1	0	1	2	2	3	3	4	5
3	-2	-2	-1	0	0	1	2	3	3	3	3	4	5
4	-2	-1	-1	0	1	2	2	3	3	3	3	4	5
5	-1	-1	0	1	2	3	3	3	3	3	4	4	5
6	-1	-1	0	1	2	3	3	3	3	3	4	5	5

표 5. 자동 학습 규칙 R_p에 해당하는 룩업 테이블
Table.5. Look-up table for self-organizing rules R_p.

\dot{e} p e p	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6
-6	-5	-5	-5	-5	-5	-4	-3	-2	-1	0	0	0	0
-5	-5	-5	-5	-4	-4	-4	-3	-2	-1	0	0	0	0
-4	-5	-5	-4	-4	-3	-3	-3	-2	-1	1	1	1	1
-3	-5	-4	-4	-3	-2	-2	-2	-2	-1	1	2	2	2
-2	-4	-4	-3	-3	-2	-2	-1	-1	0	2	2	2	3
-1	-5	-4	-3	-3	-2	-1	0	1	2	2	2	3	4
0	-4	-4	-3	-2	-2	-1	0	1	2	2	3	4	4
1	-4	-3	-3	-2	-2	-1	0	1	2	3	3	4	5
2	-3	-3	-2	-2	0	1	1	2	2	3	3	4	5
3	-2	-2	-2	-1	1	2	2	2	3	3	4	4	5
4	-1	-1	-1	-1	1	2	3	3	3	4	4	5	5
5	0	0	0	0	1	2	2	4	4	5	5	5	5
6	0	0	0	0	1	2	3	5	5	5	5	5	5

에 의해 λ 를 가변 시켰을 때의 학습 횟수에 따른 SOFC의 성능지수 IAE를 볼수있다. 그림8에서 학습이 진행됨에 따라 성능 지수 IAE는 λ 가 0.5 일때 6.4와 7.1 사이에서 진동하고 λ 가 0.1 일때 34번째 학습 이후에 6.2654로 수렴함을 볼수있다. 그림9는 λ 를 조

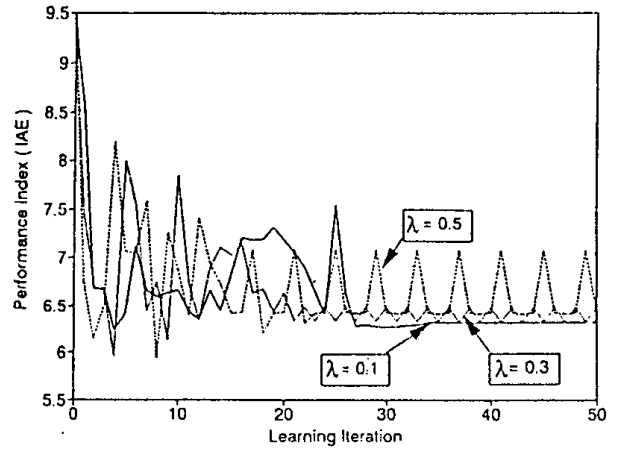


그림 8. λ 를 고정시켰을 경우 학습 횟수에 따른 성능지수 IAE의 변화 상태
Fig. 8. IAE index values of SOFC with respect to iteration number for self-organizing action when constant λ is employed.

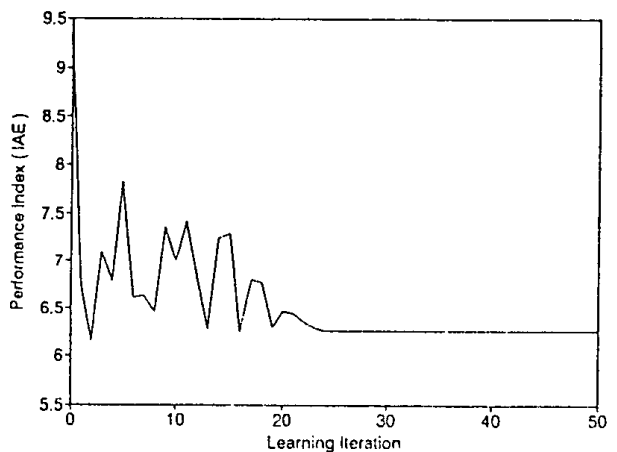


그림 9. λ 조정 알고리즘을 적용했을 경우 학습횟수에 따른 성능지수 IAE의 변화 상태
Fig. 9. IAE index values of SOFC with respect to iteration number for self-organizing action when λ modification rule is applied.

정하는 알고리즘을 이용 했을때의 IAE의 학습 횟수에 따른 변화 상태를 보여주는 그림이며 24번째 학습 이후 IAE가 6.2577로 수렴함을 볼 수 있다. 따라서 λ 를 수정하는 알고리즘을 사용한 SOFC가 λ 를 고정시킨 SOFC에 비해 수렴 속도가 빨라졌다는 것을 알 수 있다. 한편 [7-9, 11]에서 제한한 SOFC를 도입, 전차의 제어에 이용했을 경우에는 계산 시간의 증가로 인해 불안정한 응답을 보였다.

그림10은 1, 5, 10, 15, 24번 학습을 수행 했을때 SOFC의 시간에 따른 출력 특성을 나타낸 것이다. 1번째 학습 수행에서 ITAE, IAE는 각각 1,4858, 9.4078

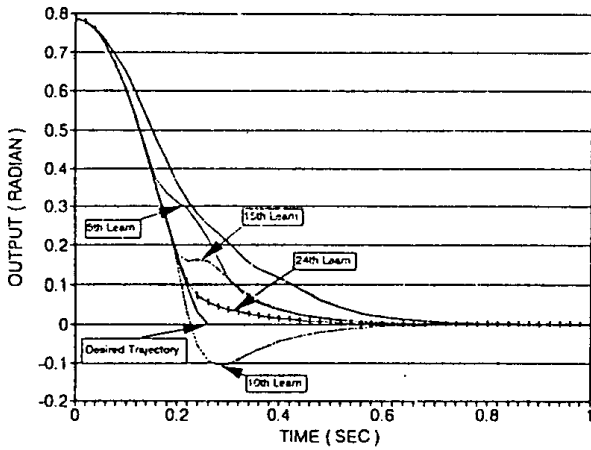
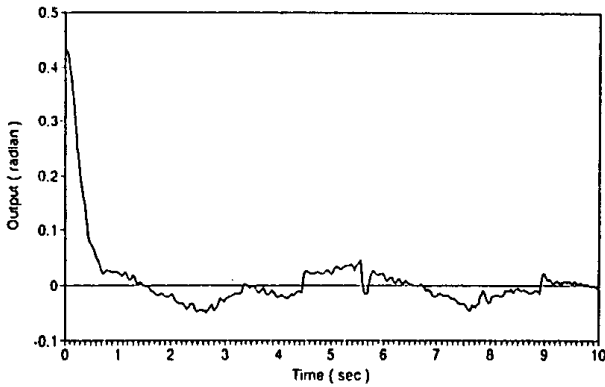


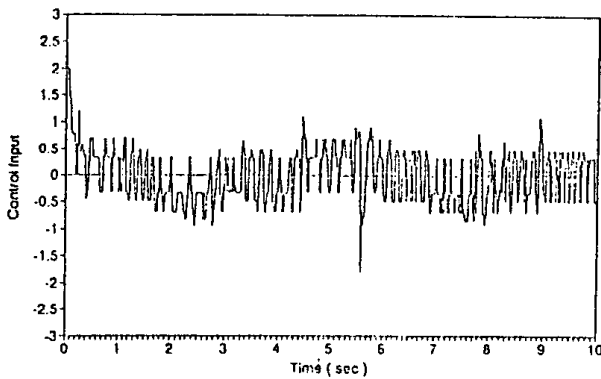
그림 10. SOFC의 출력 특성
Fig. 10. Output performance for SOFC with respect to time.

이고 학습이 완결된 24번째의 수행에서는 ITAE, IAE는 각각 0.5104, 6.2577로 감소했다. 또한 그림 10으로부터 24번째 수행에서의 출력 특성이 바람직한 출력 특성과 거의 일치함을 볼 수 있다. 학습이 진행되는 동안 도달 시간은 0.7sec에서 0.5sec로 감소했고 과도 응답은 거의 나타나지 않았음을 알 수 있다. 따라서 그림 10으로부터 SOFC의 LUT는(11, 12)식에 의한 학습 알고리즘에 의해 성능이 개선되는 방향으로 변화함을 알 수 있다.

그림 11과 그림 12는 각각 [14]에서 제안한 이원화된 제어규칙을 이용했을 때와 본 논문에서 제안한 SOFC를 이용했을 때의 실험 결과이다. 그림 11에서 출력이 -0.5 rad과 0.5 rad 사이의 구간에서 진동함을 볼 수 있는데 이것은 정밀 제어를 수행하기 위해 목표 지점으로 접근했을 때 언어 변수값의 폭을 좁게 했기 때문에 나타나는 현상이다. 그림 12에서는 (18)식과 같이 목



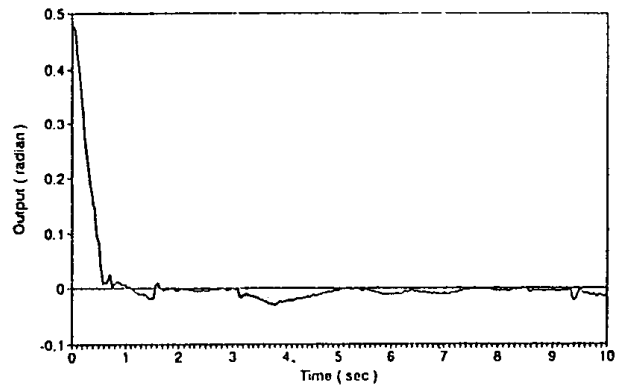
(a)



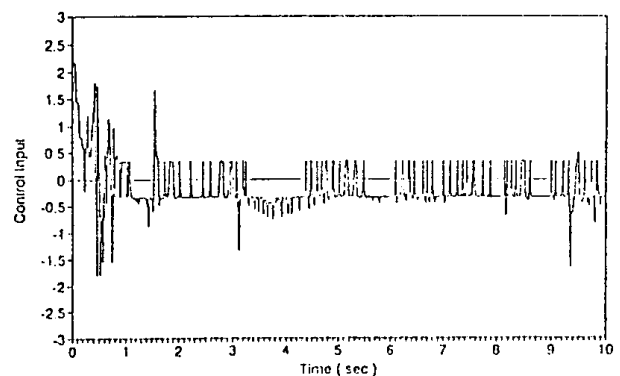
(b)

그림 11. [14]에서 제안한 이원화된 제어 규칙을 이용한 실험 결과
(a) 시간 응답
(b) 제어 입력

Fig. 11. Experimental result using FLC proposed in [14].
(a) Output performance with respect to time,
(b) Control input.



(a)



(b)

그림 12. 제안된 SOFC를 이용한 실험 결과
(a) 시간 응답
(b) 제어 입력

Fig. 12. Experimental result using SOFC.
(a) Output performance with respect to time,
(b) Control input.

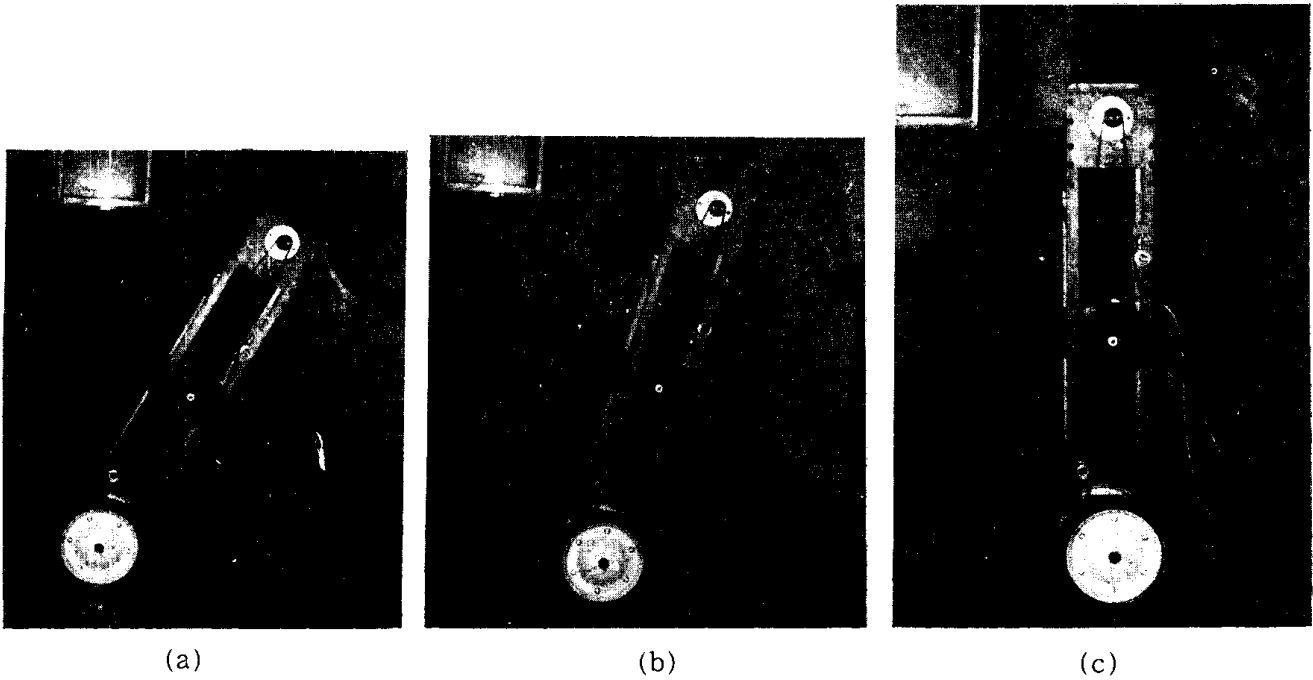


그림 13. 제안된 SOFC를 사용한 도립 진자를 제어하는 모습

Fig. 13. Experimental show-up of the control of inverted pendulum system using SOFPD algorithm.

표 지점 부근에서 FLC를 (19)식에 의해 게인값들이 조정되는 비선형적인 제어 알고리즘으로 대체했기 때문에 출력의 진동 현상을 거의 볼 수 없다. 더우기도 달 시간은 본 논문에서 제안한 SOFC를 이용했을 때 [14]에서 제안한 알고리즘을 사용했을 때 보다 1.2 sec에서 0.6 sec로 감소했다. 따라서 그림 11과 그림 12를 통해 FLC가 정밀 제어에 적합하지 않으며 이를 보완하기 위해서는 목표지점 부근에서 FLC를 PD 제어기로 대체해야 한다는 결론을 얻을 수 있다.

그림 12는 본 논문에서 제안한 SOFC를 이용했을 때 도립 진자가 제어 되는 모습을 보여주는 것이다. 이 그림으로 부터 불확실하고 비선형적인 특성을 갖는 도립 진자의 실시간 제어에 제안된 SOFC가 성공적으로 적용됨을 알 수 있다.

V. 결 론

본 논문에서는 퍼지제어 알고리즘이 불안정한 시스템을 정밀 제어 하는데 부적합하다는 것을 보이고, FLC와 PD제어의 상호 보완적인 측면을 제시 이들도립 진자의 제어 실험을 통해 입증하였다. 또한 퍼지 제어기의 성능을 좌우하는 이득 상수들에 대한 체계적인 조정 방법을 제시 하였으며 시 불편 목표값에 대해 점근적으로 수렴하고 수렴속도를 향상시킨 자동 학습 퍼지 제어기를 구현하였다.

자동 학습 퍼지 제어기는 스스로 제어 성능을 향상 시킴으로써 반복 작업을 통해 제어 규칙을 수정해야 하는 수고를 덜어줄 수 있었다. 그러나 SOFC에는 시행 오차를 통해 조정해야 하는 파라미터들이 많이 있다. 따라서 파라미터들과 출력 특성간의 분석을 통해 이들을 자동 조정할 수 있는 알고리즘에 대한 연구가 필요하다. 또한 SOFC의 안정도에 대한 수학적 분석이 필요하다.

参 考 文 献

- [1] L. A. Zadeh, "Fuzzy 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, "Fuzzy 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," *Automatica*, 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," *Automatica*, 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 cesign of fuzzy control systems," *Proc. Int. Conf. on Fuzzy Logic & Neural Networks*, pp. 405-408, 1990.

著 者 紹 介

李 鏞 魯(正會員)

1967年 10月 8日生. 1990年 2月 한양대학교 전자공학과 졸업. 1992年 2月 한양대 전자공학과 졸업(공학석사). 현재 현대자동차 기술연구소 재직중. 주관심 분야는 퍼지시스템, 신경회로망, 마이크로프로세서 응용 등임.



金 兌 垣(正會員)

1965年 12月 28日生. 1988年 2月 한양대학교 전기공학과 졸업. 1990年 2月 한양대 대학원 전기공학과 졸업(공학석사). 1990年 3月~현재 한양대 대학원 전자공학과 박사과정 재학중. 주관심분야는 로봇트

비전, 뉴로퍼지 등임.

徐 一 弘(正會員) 第28卷 A編 第8號 參照

현재 한양대학교 전자공학과 부교수.