

## 규칙기반 자기학습 제어기의 알고리즘 개선과 응용에 관한 연구

正會員 姜 信 出\*, 林 永 道\*\*

### A Study on the Algorithm Improvement and Application of the Rule-based Self-Organizing Controller

Shin Chul Kang\*, Young Do Lim\*\* Regular Members

#### 要 約

규칙기반 자기학습 제어기의 알고리즘을 수정하여 성능을 향상시켰다. 제어기 설계법은 성능평가 항목에 근거하여 기준모형을 선정하였다. 제어오차를 감소시키기 위하여 연속변수를 가지는 제어기와 heuristic knowledge에 의한 방법으로 규칙을 수정하였으며, 비례적분미분형 규칙기반 자기학습 제어기의 도입으로 수렴특성을 개선하였다. 빠른 응답특성을 얻기위하여 제어 파라미터인 입,출력 이득을 실험적으로 결정하여 적절하게 선택하였다. 모의실험으로 성능평가 항목을 조사하여 수정 전,후의 규칙기반 자기학습 제어기의 출력응답 특성을 비교 고찰하였다. 실험의 결과로서  $\mu$ -컨트롤러를 이용하여 직류 서어보 모터의 속도제어의 실시간 제어를 위한 실험장치를 구현하였다.

#### ABSTRACT

The performance of the rule-based self-organizing controller was improved by modifying the algorithm. The controller was designed by choosing the reference model based on the items of performance evaluation. In order to reduce the control error, the rule was modified according to the method of heuristic knowledge and the controller with the continuous parameters. The convergence characteristics was also improved by applying Proportional-Integral-differential(PID) algorithm to the rule-based self-organizing controller. The proper choice of input/output gain, that is, the control parameters which were determined by experiments enhanced the response characteristics. The response characteristics of the rule-based self-organizing controller before and after the modifications were compared with the items for performance evaluated by computer simulations. The experimental equipments were implemented to perform real-time control of DC servo motor using  $\mu$ -controller as the results of experiments.

\*韓國通信 釜山海底 中繼局  
KT Busan Cable Landing Station

\*\*東亞大學校 電子工學科  
Dept. of Electronic Engineering, Dong-A Univ.  
論文番號 : 94255-0924  
接受日字 : 1994年 9月 24日

[ . 서 론

퍼지이론은 애매하고 불확실한 정보로부터 유용한 정보를 얻기위한 수학적 이론으로 1965년 자데(Zadeh) 교수가 제안한 이래,<sup>[1]</sup> 최근 몇년 사이에 많은 산업응용 분야에서 활발히 이용되고 있다. 퍼지제어는 다변위 공정의 제어 시스템에서 모델을 기준으로 한 기존제어기의 부적합한 점을 보완 대체하고, 빠른 처리속도와 강인성을 바탕으로 기존의 제어 이론으로 제어하기 어려운 대상들을 성공적으로 제어하여 왔으나, 한정된 제어 규칙을 사용하기 때문에 시스템 매개변수가 변할 경우 제어 규칙을 재구성하는 문제점이 발생된다.<sup>[2]</sup> 이를 보완하기 위한 방법으로 퍼지이론을 이용하여 퍼지변수들에 주어진 소속함수들을 스스로 학습 조정할 수 있는 자기 학습 제어기(Self Organizing Controller : SOC)가 T.J. Procyk, Yamazaki 등에<sup>[3]</sup> 의해 시도되었고, John yen<sup>[4],[5]</sup> 등에 의해 단일력 단출력(Single Input Single output : SISO)뿐만 아니라, 다변수 시스템에서도 응용됨을 입증하였다. 그러나 기존의 SOC는 정상 상태에서 제어오차가 크고, 설정치부근에서 수렴특성이

나쁜점과 응답속도가 느리다는 문제점이 있다. 따라서 본 논문에서는 연속변수를 가지는 제어기와 경험적 발견 지식(Heuristic Knowledge)에 의한 규칙 수정방법을 이용하여 정상상태에서 제어오차를 감소시키고, 2차원 성능지수 값을 상호보완한 비례적분미분(PID)형 SOC로 3차원 성능지수(Performance Index : PI)값을 구하여 수렴특성을 개선하였으며, 3차원 성능지수로 부터 얻어진 기준모델의 제동계수와 관련된 제어파라미터인 임,출력 이득을 실험적으로 적절하게 선택하여 응답특성을 개선하였다. 성능평가 항목으로는 평균 절대치 오차(Integral of the Absolute magnitude of the Error : IAE), 상승시간, 정착시간 그리고 과도현상등을 조사하여 최적의 학습수 만큼 학습시켜 생성, 수정된 규칙상태에서 임의의 프로세스를 제어하였다. 제안된 수정 SOC 알고리즘으로 기존의 SOC와 출력응답 특성을 모의실험에서 비교 고찰함으로써 수정된 SOC 알고리즘을 가진 퍼지제어기의 우수함을 입증하였으며, 하드웨어 구현으로 16 bit  $\mu$ -controller 인 80196과 전력 스위칭 소자 IGBT를 이용하여 직류서어보 모터의 속도제어의 유용성을 확인하였다.

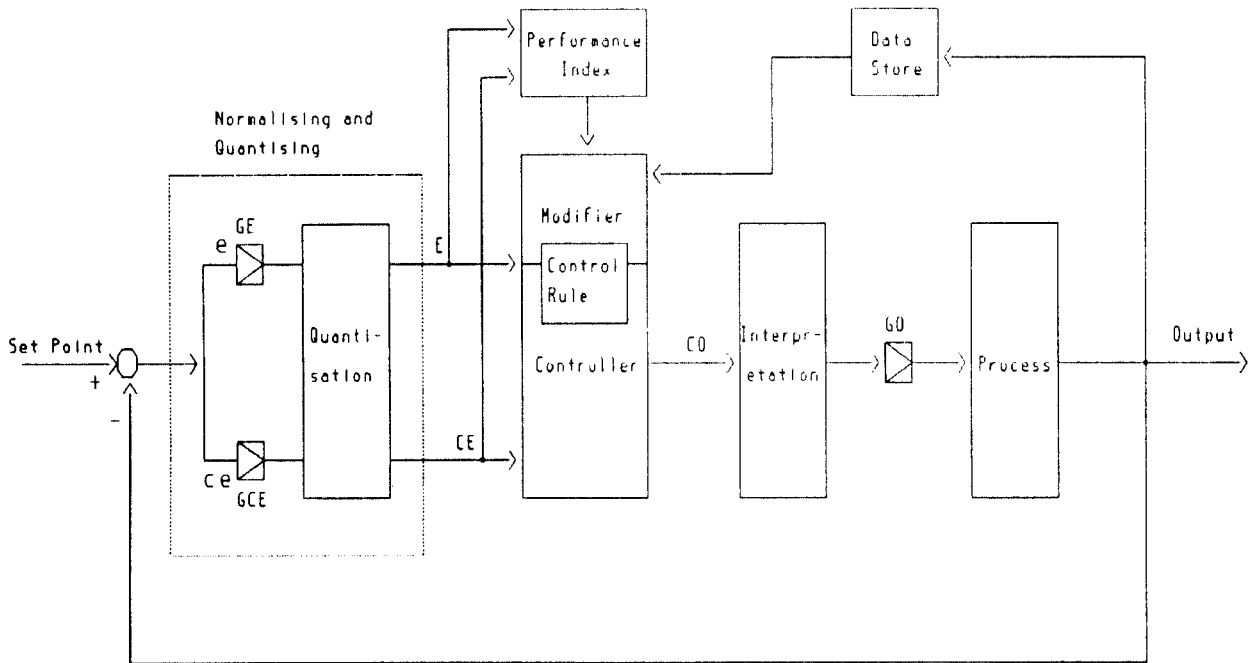


그림 1. SOC의 기본계통도  
Fig. 1. Block diagram of the SOC algorithm

## II. 자기학습제어기의 일반적 구조

일반적인 자기학습 제어기는 규칙을 스스로 학습할 수 있는 규칙기반 학습제어기로서 규칙 획득은 성능지수, 규칙수정 알고리즘과 모델로 구성되며, 그림 1 에서 SOC의 기본 계통도를 나타내었다. 입력변수는 설정치 R과 오차, 그리고 오차의 변화로 하고 출력변수는 프로세스의 출력과 제어출력의 변화로 하였다.

### 1. 양자화

SOC 알고리즘에 있어서 퍼지 변수가 다음과 같다고 하면, 전건부(제어상황:제어기입력) : 오차(error:e), 오차의변화(change in error:ce) 후건부(제어성질:제어기 출력):제어출력의 변화 (Change in output:co) 이러한 퍼지변수에 대한 전체집합은 식(1)과 같이 이산 집합(Discrete set)으로 표현할 수 있고 정규화된다.

$$\begin{aligned}
 E &= \{e\} \cong \{-6, -5 \dots -1, 0, +1 \dots +5, +6\} \\
 CE &= \{ce\} \cong \{-6, -5 \dots -1, 0, +1 \dots +5, +6\} \\
 CO &= \{co\} \cong \{-6, -5 \dots -1, 0, +1 \dots +5, +6\}
 \end{aligned}
 \tag{1}$$

따라서 측정값의 영역과 정규화 값의 영역사이에 사상 연산(Mapping Operation)을 SOC에 도입하고, 정규

화는 환산계수(Scaling)와 양자화의 두가지 과정으로 구성하였으며 환산과정(Scaling Procedure)은 식(2)로 나타내었다.

$$\begin{aligned}
 eg &= e \times GE \\
 ceg &= ce \times GCE \\
 cog &= co \times GO
 \end{aligned}
 \tag{2}$$

여기서, GE와 GCE는 입력이득, GO 는 출력이득, eg 와 ceg는 측정된 입력값 e와 ce, 그리고 cog는 측정된 co 에 대한 환산값(Scaled value)이다. 또한 환산값 eg, ceg와 cog는 식(1)에서 정의한 E와 CE, CO의 전체집합에 대한 요소들로 양자화하였다. 양자화의 범위는 제어환경에 따라 선택하고 이산집합을 이용하면 식(3)과 같이 양자화의 표현을 간략화 할 수 있다.

$$\begin{aligned}
 e^* &= (eg)_q \\
 ce^* &= (ceg)_q \\
 co^* &= (cog)_q
 \end{aligned}
 \tag{3}$$

여기서, q는 양자화 표시.

### 2. 성능지수

표 1. Yamazaki가 제안한 성능지수  
Table 1. Performance index proposed by Yamazaki

e\ce*	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6
-6	-6	-6	-6	-6	-6	-6	-6	-4	-3	-2	-1	0	0
-5	-6	-6	-6	-5	-5	-4	-4	-3	-2	-1	0	0	0
-4	-6	-6	-5	-5	-4	-3	-3	-2	-1	0	0	0	0
-3	-6	-5	-5	-4	-3	-2	-2	-1	0	0	0	0	1
-2	-6	-5	-4	-3	-2	-1	-1	0	0	0	0	1	2
-1	-5	-4	-3	-2	-1	-1	-1	0	0	0	1	2	3
0	-4	-3	-2	-1	-1	0	0	0	1	1	2	3	4
1	-3	-2	-1	0	0	0	1	1	1	2	3	4	5
2	-2	-1	0	0	0	0	1	1	2	3	4	5	6
3	-1	0	0	0	0	1	2	2	3	4	5	5	6
4	0	0	0	0	1	2	3	3	4	5	5	6	6
5	0	0	0	1	2	3	4	4	5	5	6	6	6
6	0	0	1	2	3	4	6	6	6	6	6	6	6

성능지수는 제어기의 성능에 대한 척도이며 프로세스가 요구하는 상태에 얼마나 근접해 가는가의 기준으로 제어기 성능의 평가와 관련된 퍼지 함의로부터 구해지고 성능지수값은 과거의 어느 샘플링 순간에서 제어기 출력의 변화로 간주할 수 있으므로 일반적으로 식(4)와 같은 제어규칙의 형태를 가진다.

$$IF e^* = 1 \text{ and } ce^* = 2 \text{ THEN } PI(t) = -3 \quad (4)$$

여기서, t는 현재시간.

PI(t) : 시간 t에 있어 성능지수

표 1은 Yamazaki가 1982년에 제안한 성능지수표로서  $e^*$ 와  $ce^*$ 의 범위는 +6에서 -6, PI는 13×13의 행렬의 값을 가지도록 하였으며  $e^*$ 가 +6으로 양의 최고값이고  $ce^*$ 가 -6일때 PI의 값은 0으로서 프로세스가 정상 상태에 도달함을 의미하며 PI의 절대값이 크면 클수록 프로세스는 정상상태를 훨씬 벗어남을 의미한다. 표 1에서 보는 바와 같이  $e^*$ 가 +6으로 양의 최고값이고  $ce^*$ 가 -6으로 음의 최고값이라면 프로세스는 동작상태에 있음을 알 수 있으며,  $ce^* = -6$ 은 오차가 급격히 감소됨을 의미한다.  $e^*$ 가 적어질수록  $ce^*$ 는 더욱 적어지며,  $e^* = ce^* = 0$ 이면 정상상태에 도달하게 된다.<sup>[7])</sup>

### 3. 규칙수정

규칙은 성능지수표에 의해 과거에 추론된 출력을 보정함으로써 수정하거나 생성되어진다. 식(5)의 규칙 수정방법은 프로세스가 현재 상태에 대하여 n번째 전 샘플링점에 있어 제어기 출력이 확실하게 변화된다는 가정으로 본 알고리즘이다.

$$R(t-nT) = u(t-nT) + PI(t) \quad (5)$$

여기서, R(t-nT) : 시간 (t-nT)에 있어 프로세스 상태의 규칙

u(t-nT) : 시간 (t-nT)에 있어 제어기 출력  
T : 샘플링 주기

n은 프로세스 시정수를 고려하여 결정되며 지연보상(DELay in reward : DEL)이라 한다. 식(5)는 한개의 샘플링점에서의 제어기 출력을 고려했을 때로서 단일 지연보상(Single DELay in reward : SDEL)이라 하며, 한개 이상의 샘플링 점에 있어 제어기 출력이 현

재 상태에서 프로세스의 상태를 처리해야 한다면 프로세스의 상태에 영향을 미치는 모든 샘플링을 검토해야 하므로 이것을 다중 지연보상(Multi DELay in reward : MDEL)이라 하고 식(6)으로 표현된다.

$$R(t-nT) = u(t-nT) + m(n) \times PI(t) \quad (6)$$

여기서, m(n) : 시간 (t-nT)에서 보상에 대한 가치.

n = 4, 5, 6 이라면 t-4T, t-5T, t-6T 에서 제어기 출력이 처리되며 m(4) = m(5) = m(6) = 1/3로 동등하게 보상되어야 한다.

## III. SOC의 수정 알고리즘

SOC의 장점은 스스로 제어규칙을 획득할 수 있고, 최적의 학습후에 좋은 응답특성을 얻을 수 있다는 것이다. 그러나 기본 SOC의 알고리즘은 정상상태에서 제어오차가 크고, 입력에 대한 수렴특성이 나빠져 응답속도가 느린 단점이 있다. 이러한 SOC 단점을 개선하기 위하여 수정알고리즘을 제안하였다.

### 1. 성능평가 항목

제어기의 성능은 설정치에 대한 응답특성으로 평가하였으며 정량적인 성능을 평가하기 위해 다음의 항목을 도입하였다.

TR [sec] : 최종치의 10%에서 90%까지의 상승시간으로서 응답의 속용성을 측정.

OV [%] : 과도응답으로서 제동특성을 측정.

TS [sec] : 정상상태에서 오차 5%로 정착되는 시간으로서 설정치에 대한 응답수렴 특성 측정.

IAE (Integral of the absolute of the error) : 절대 편차 적분으로서 제어를 시작하여 시스템이 안정된 상태에 이르기 까지 발생된 오차의 합으로서 프로세스에 있어 정상상태의 제어오차를 측정.

### 2. 제어오차 감소

제어오차를 감소시키기 위하여 연속변수를 가지는 제어기와 Heuristic Knowledge에 의한 방법으로 규칙수정을 하였다.

2.1 연속 변수를 가지는 제어기

이산적인 값에서 양자화는 오차를 허용할 뿐만 아니라 전체적인 성능을 나쁘게 한다. 이러한 양자화의 단점은 양자화의 레벨을 세분화 시킴으로써 개선 향상 될 수 있지만 계산시간과 데이터 메모리가 증가한다. 그러므로 최적의 방법은 이산적인 변수에 가중치를 취하여 연속적인 변수를 만드는 것으로서 완전한 연속변수는 될 수 없지만 연속변수를 가지는 제어기와 거의 같은 효과를 나타낼 수 있다. 비례적분(Proportional Integral : PI)형 제어기에 대해서  $eg = 0.3$ 이고  $ceg = 0.6$ 인 경우, 4개의 다른 영향대의 소속함수는 식(7)로 나타내어진다.

- $\mu(0,0) : e^* = 0$  이고  $ce^* = 0$ 에 대한 영향대
- $\mu(0,1) : e^* = 0$  이고  $ce^* = 1$ 에 대한 영향대
- $\mu(1,0) : e^* = 1$  이고  $ce^* = 0$ 에 대한 영향대 (7)
- $\mu(1,1) : e^* = 1$  이고  $ce^* = 1$ 에 대한 영향대

표 2. 연속입력에 대한 계산값  
Table. 2. Calculation for continuous input

c* we	ce* wce	0	1
		0.4	0.6
0	0.7	w(0, 0) 0.28	w(0, 1) 0.42
1	0.3	w(1, 0) 0.12	w(1, 1) 0.18

A. Weight for each function

A. 각 함수에 대한 가중치

e*	ce*						
	-2	-1	0	1	2	3	
	-2	0.09	0.21	0.3	0.21	0.09	0.0
	-1	0.21	0.49	0.7	0.49	0.21	0.0
	0	0.3	0.7	1.0	0.7	0.3	0.0
1	0.21	0.49	0.7	0.49	0.21	0.0	
2	0.09	0.21	0.3	0.21	0.09	0.0	
3	0.0	0.0	0.0	0.0	0.0	0.0	

B.  $\mu(0,0)$

e*	ce*						
	-2	-1	0	1	2	3	
	-2	0.0	0.09	0.21	0.3	0.21	0.09
	-1	0.0	0.21	0.49	0.7	0.49	0.21
	0	0.0	0.3	0.7	1.0	0.7	0.3
1	0.0	0.21	0.49	0.7	0.49	0.21	
2	0.0	0.09	0.21	0.3	0.21	0.09	
3	0.0	0.0	0.0	0.0	0.0	0.0	

C.  $\mu(0,1)$

e*	ce*						
	-2	-1	0	1	2	3	
	-2	0.0	0.0	0.0	0.0	0.0	0.0
	-1	0.09	0.21	0.3	0.21	0.09	0.0
	0	0.21	0.49	0.7	0.49	0.21	0.0
1	0.3	0.7	1.0	0.7	0.3	0.0	
2	0.21	0.49	0.7	0.49	0.21	0.0	
3	0.09	0.21	0.3	0.21	0.09	0.0	

D.  $\mu(1,0)$

e*	ce*						
	-2	-1	0	1	2	3	
	-2	0.0	0.0	0.0	0.0	0.0	0.0
	-1	0.0	0.09	0.21	0.3	0.21	0.09
	0	0.0	0.21	0.49	0.7	0.49	0.21
1	0.0	0.3	0.7	1.0	0.7	0.3	
2	0.0	0.21	0.49	0.7	0.49	0.21	
3	0.0	0.09	0.21	0.3	0.21	0.09	

E.  $\mu(1,1)$

e*	ce*						
	-2	-1	0	1	2	3	
	2	0.0252	0.0966	0.1722	0.1848	0.1134	0.0378
	-1	0.0696	0.2668	0.4756	0.5104	0.3132	0.1044
	0	0.1092	0.4186	0.7462	0.8008	0.4914	0.1638
1	0.0948	0.3634	0.6478	0.6952	0.4266	0.1422	
2	0.0524	0.1932	0.3444	0.3696	0.2268	0.0756	
3	0.0108	0.0414	0.0738	0.0792	0.0486	0.0162	

F.  $\mu(0,3,0,6)$

소속함수에 대한 가중치는 식(8)로 나타낸다.

- $w(0,0) = we(0) \times wce(0) : \mu(0,0)$ 에 대한 가중치
- $w(0,1) = we(0) \times wce(1) : \mu(0,1)$ 에 대한 가중치
- $w(1,0) = we(1) \times wce(0) : \mu(1,0)$ 에 대한 가중치 (8)
- $w(1,1) = we(1) \times wce(1) : \mu(1,1)$ 에 대한 가중치

여기서,  $we(i) : e^* = i$ 에 대한 가중치

$wce(i) : ce^* = i$ 에 대한 가중치.

식(7)과 식(8)에 의거 연속적인 소속함수의 값은 식(9)로 된다.

$$\begin{aligned} \mu(0,3,0,6) &= \mu(0,0) \times w(0,0) + \mu(0,1) \times w(0,1) \\ &\quad + \mu(1,0) \times w(1,0) + \mu(1,1) \times w(1,1) \\ &= \mu(0,0) \times 0.28 + \mu(0,1) \times 0.42 \quad (9) \\ &= \mu(1,0) \times 0.12 + \mu(1,1) \times 0.18 \end{aligned}$$

표 2는 연속 입력  $eg = 0.3$ ,  $ceg = 0.6$  에 대한 각 함수의 가중치 값을 계산한 소속정도이다. 표 2와 같이 제어기의 연속입력에 대한 영향대가 구해지면 규칙표와 영향대로부터 제어기의 연속적인 출력  $du$ 는 식(10)으로 구할 수 있다.

$$du = \frac{\sum_j \mu(e_i^*, ce_j^*) \times R(e_i^*, ce_j^*)}{(i \times j) - k} \quad (10)$$

여기서,  $k$ 는 규칙표에서 영향대 안의 요소가 0인 개수.

연속 알고리즘에 있어서는 제어기 출력뿐만 아니라 학습도 역시 연속적이어야 하므로 성능지수에 대한 값은 성능지수표 4개 요소의 가중치 합으로 계산된다. 만일  $eg = 0.3$ 이고  $ceg = 0.6$ 이라면,  $PI(t)$ 의 값은 식(11)로 구해진다.

$$PI(t) = \sum_{i=0}^1 \sum_{j=0}^1 \{PI(i, j) \times w(i, j)\} \\ = PI(0, 0) \times w(0, 0) + PI(0, 1) \times w(0, 1) \\ + PI(1, 0) \times w(1, 0) + PI(1, 1) \times w(1, 1) \quad (11)$$

또한  $DEL = n$ 이고  $eg(t-nT) = 3.4$ ,  $ceg(t-nT) = 2.7$ 이면 식(12)의 방법으로 규칙을 수정한다.

$$R(3, 2)(t-nT) = u(t-nT) + w(3, 2)(t-nT) \times PI(t) \\ R(3, 3)(t-nT) = u(t-nT) + w(3, 3)(t-nT) \times PI(t) \\ R(4, 2)(t-nT) = u(t-nT) + w(4, 2)(t-nT) \times PI(t) \quad (12) \\ R(4, 3)(t-nT) = u(t-nT) + w(4, 3)(t-nT) \times PI(t)$$

여기서,  $R(i, j)(t-nT)$  :  $(t-nT)$ 에 있어  $e^* = i$ ,  $ce^* = j$ 의 규칙

$u(t-nT)$  :  $(t-nT)$ 에 있어 제어기 출력

$w(i, j)(t-nT)$  :  $(t-nT)$ 에 있어  $e^* = i$ ,  $ce^* = j$ 에 대한 가중치.

### 2.2 Heuristic knowledge에 의한 규칙수정

SOC알고리즘에서 있어 비례적분 제어의 특성을 바탕으로 Heuristic knowledge에 의한 규칙수정 방법을 택하여 정상상태에서의 제어오차를 감소시켰다.

표 3은 수정전 이산 알고리즘에 의해 생성된 규칙행렬로서 점선영역은 프로세스가 수렴하는데 필요한  $e^* = 0$ ,  $ce^* = 0$ 에 대한 영향대이다. 합성규칙을 통해 추론된 실제출력은 최종제어기 출력에서의 변화로서  $e^* =$

표 3. 수정전 알고리즘에 의해 생성된 규칙  
Table 3. Rule made by unmodified algorithm

$e^* \backslash ce^*$	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6
-6	*	*	*	60	*	-60	-60	-60	-60	*	*	*	*
-5	*	*	-60	*	*	*	*	*	*	*	*	*	*
-4	*	-60	*	*	*	*	*	*	*	*	*	*	*
-3	-60	*	*	*	*	-60	*	*	*	*	*	*	*
-2	-60	*	*	-60	-60	*	*	*	*	*	10	*	*
-1	*	*	*	-60	-50	-30	-20	0	10	*	30	*	*
0	-60	*	-50	-50	-50	*	10	20	40	*	60	*	*
1	*	*	-50	-40	-40	*	20	40	50	*	60	*	*
2	-50	*	-50	*	*	*	*	60	50	60	*	*	*
3	*	*	*	*	*	*	*	*	60	60	*	*	*
4	-20	*	*	*	*	*	60	60	*	*	*	*	*
5	*	*	*	*	*	*	*	*	*	*	*	*	*
6	*	*	60	60	60	60	60	*	*	*	*	*	*

$ce^* = 0$  일 때의 값이 0.37이 된다. 출력 양자화가 존재할 때 이값은 0의 값으로 양자화가 되어 어떠한 제어오차도 발생하지 않는다. 그러나, 연속 알고리즘에 있어서는 정상상태에서의 추론된 출력이 항상 존재하므로 여전히 제어오차가 발생시킨다. 따라서 아래사항을 만족하면 제어오차를 감소시킬 수 있으며 제어규칙에 대하여 Heuristic knowledge라 한다.

첫째,  $e^* = ce^* = 0$ 에 대한 규칙행렬의 요소  $R(0, 0) = 0$

둘째, 규칙이  $e^* = ce^* = 0$ 일 때의 영향대 안에서 수정될 때 수정전 규칙이  $R_p(i, j)$ 라 하고, 수정후 규칙을  $R_1(i, j)$ 라 할 때, 새로운 규칙은  $R_n(i, j) = (R_p + R_1)/2$ 로 하고 대칭위치에 대하여는  $R(-i, -j) = -R_n$ 로 한다.

셋째,  $R_1(i(0), j(0)) < 0$ ,  $R_1(i(0), j(0)) > 0$  일 때  $R_n = 0$ 으로 둔다.

### 3. 수렴특성 개선

정상상태에서 수렴특성을 개선키 위한 방법으로서 비례적분 제어보다 비례적분미분(PID) 제어의 수렴특성이 양호한 장점을 이용하였다. PID 제어의 알고리즘은

$$u = K_p \cdot e + (1/T_i) \cdot \int e \cdot dt + T_d \cdot (d/dt) \cdot e \quad (13)$$

으로 표현되며, 식 (13)을 이산시간 중분 알고리즘으로 나타내면

$$\begin{aligned} du &= K_p \cdot ce + (1/T_i) \cdot e + T_d \cdot cce \\ &= ce^* + e^* + cce^* \end{aligned} \quad (14)$$

가 된다.

여기서,  $cce = \text{Change in change in error}$ ,  $du = \text{Change in output}$  이다.

이와같이 식 (14)에서 비례, 적분, 미분항을 퍼지 논리에 적용하면 식 (15)의 3차원 공간형태의 제어 규칙표와 성능 지수표를 생성시킬수 있다.

$$\begin{aligned} Ri: & \text{IF } e^* = Ei \text{ and } ce^* = CEi \text{ and } \\ & cce^* = CCEi \text{ THEN } du = U_i \\ Pli: & \text{IF } e^* = Ei \text{ and } ce^* = CEi \text{ and } \\ & cce^* = CCEi \text{ THEN } PI = PI_i \end{aligned} \quad (15)$$

여기서,  $cce^* = cce \times GCCE$

$GCCE$  :  $cce$ 에 대한 입력이득.

그러나, 이와같이 수정전 2차원의 비례적분형 행렬대

신 PID형 3차원의 공간행렬 형태를 취하면 수렴특성은 양호하게 되지만, 계산시간, 메모리 용량 그리고 컴퓨터 하드웨어 속도 등의 문제로 인하여 실현시키는 것은 복잡하고 난해하다. 따라서 본 논문에서는 전문가의 경험적 지식과 시행착오의 반복으로 얻어진 2차원의 성능지수 PI(2-D)와 2차원 제어출력 Output(2-D)의 요소에 PID의 미분항인  $cce^*$ 의 값을 보완한 식 (16)의 PID형태의 SOC방법을 취하여 수렴특성을 개선하였다.

$$\begin{aligned} PI &= PI(2-D) + cce^* \\ \text{Output} &= \text{Output}(2-D) + cce^* \end{aligned} \quad (16)$$

### 4. 응답속도

빠른 응답특성을 얻기 위해서는  $e$ 가 클 때 모델의 시정수를 적게 하여 GCE를 적게 선정하게 하고,  $e$ 가 작을 때는 모델의 시정수를 크게하여 서서히 수렴하도록 GCE를 크게 선정한다. 이것은 프로세스의 오차가 크면 빨리 상승하고 오차가 적으면 서서히 수렴함을 뜻한다. 오차를 적게 하기위하여 양자화를 세밀하게 할 필요는 있으나, 프로세스가 필요로 하는 것은 빠르게 설정치에 도달하는 것이므로 오차의 크기를 조정하였다. 그림 2에 나타난 것처럼 비선형 양자화를 이용하여 0에서 6까지 eg의 전체 범위를 0과 100사이로  $e^*$ 를 사상시켜 오차가 작으면 양자화를 세밀하게 하고 오차가 크면 양자화의 범위는 넓어지도록 하였다. 식(17)은  $e^* = i$ 로 사상되어진 eg의 값을  $e_i$ 라 할 때의 양자화이다.

$$Qe(i) = e_i \quad (17)$$

이것을 선형 양자화시키면 식(18)이 되며

$$\begin{aligned} Qe &= \{Qe(0), Qe(1), Qe(2), Qe(3), Qe(4), Qe(5), Qe(6)\} \\ &= (0, 1, 2, 3, 4, 5, 6) \end{aligned} \quad (18)$$

식(18)에 대응하는 비선형 양자화는 식(19)로 표현할 수 있다.

$$Qe = (0, 1, 3, 6, 10, 15, 100) \quad (19)$$

이러한 양자화의 형태는 인간의 지각을 모방하였다고 할 수 있다. 양자화 값의 범위는 99와 100 사이의 값보다 0과 1 사이에서 더욱 예민한 반응을 보인다. 비선형 양자화의 장점은 양자화 레벨이 선형 양자화와 동일해지

며, 오차가 크면 제어가 평활되지 않는 단점이 있다.

$Qe(0) = Qce(0) = 0$ 와  $Qe(6) = Qce(6) = 100$ 으로 동 특성범위가 고정되고 수정전 선형 양자화와 같이 설정치 부근에서 정확하게 접근하도록  $Qe(1) = Qce(1) = 1$ 로 한다면,  $Qe(2)$ 에서  $Qe(5)$ 까지, 그리고  $Qce(2)$ 에서  $Qce(5)$ 의 값은 어떻게 택할 것인가가 관건이 된다. 따라서 빠른 응답이 필요하면 큰 값을 택하도록 하고, 평활한 제어에서는 아주 작은 값을 택하였다. 이와같은 방법으로 표 4의 비선형 양자화 성능지수표를 구하여 세밀하고 빠른 응답속도를 얻도록 하였다.

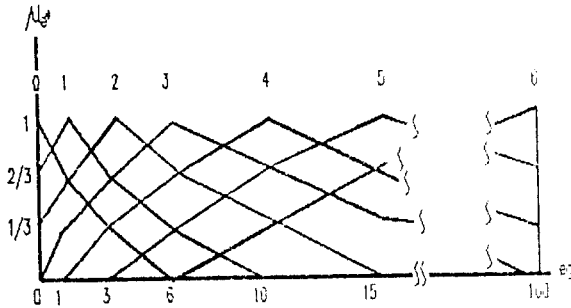


그림 2. 비선형 양자화  
Fig. 2. Nonlinear quantization

#### IV. 제어기의 설계법

지금까지 SOC의 성능을 개선하기 위해 알고리즘의 다양한 수정을 행하였다. 본 장에서는 제어 파라미터를 기준으로 한 제어기의 입, 출력 변수와 샘플링 주기의 정규화에 대해 논하고 제어기의 설계에 적용할 프로세스를 해석하여 기준모델 적응제어기(MRAC)의 기능을 갖는 SOC의 기준모델을 선정하였다.

##### 1. 제어기의 정규화

정규화된 제어기 출력  $u$ 와 프로세스 출력  $Y$ 의 범위는  $[-100, 100]$ , 정규화된 샘플링 주기  $T_{sn}=1$ 로 하였다. 그러나 실제 응용에 있어 이들 변수의 값은 다양한 범위를 취하게 된다. 직류서보 모터의 속도제어에 있어서 프로세스 출력인 속도  $Y_p$ 의 범위를  $0 \sim Y_{pmax} [rpm]$ , 프로세스 입력  $u_p$ 의 범위는  $0 \sim u_{pmax} [V]$ , 샘플링주기  $T_s$ 를  $t_s [ms]$ , 그리고 프로세스의 시정수  $T_p$ 는  $t_p [ms]$ 라 할 때, 정규화된 제어기 시스템의 제어파라미터인 GE, GCE, GCCE 및 GO는 재 조정되어야 하는 문제점이 있으므로, 실제 프로세스는 그림 3과 같이 정규화하였다. 정규화된 샘플링 주기  $T_{sn}$ 을 1로 할 때, 프로세스의 정규화된 시정수

표 4. 비선형 양자화 성능지수  
Table 4. Performance index with nonlinear quantization

$e \cdot ce^*$	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6
-6	-100	-100	-100	-100	-100	-100	-100	-98.0	-95.0	-91.3	-86.5	0.0	0.0
-5	-100	-28.4	-21.9	-17.8	-15.2	-13.6	-12.6	-11.6	-9.4	-6.6	0.0	0.0	0.0
-4	-100	-22.8	-16.3	-12.2	-9.6	-8.0	-7.0	-6.0	-4.1	0.0	0.0	0.0	0.0
-3	-100	-19.3	-12.8	-8.7	-6.1	-4.3	-3.5	-2.5	0.0	0.0	0.0	4.4	51.5
-2	-100	-17.1	-10.6	-6.5	-3.9	-2.3	-1.5	-0.5	0.0	0.0	2.9	6.6	57.0
-1	-100	-14.7	-8.6	-4.0	-7.8	-1.0	-0.3	0.0	0.4	2.1	5.3	10.6	74.3
0	-100	-12.6	-7.0	-3.5	-1.3	-0.5	0.0	0.5	1.3	3.5	7.0	12.6	100
1	-74.5	-10.6	-3.3	-2.1	-0.4	0.0	0.3	1.0	2.3	4.8	8.6	14.7	100
2	-57.0	-6.6	-2.9	0.0	0.0	0.5	1.5	2.5	3.9	6.5	10.6	17.1	100
3	-51.5	-4.4	0.0	0.0	0.0	2.5	3.5	4.5	6.1	8.7	12.8	19.3	100
4	-46.4	0.0	0.0	0.0	4.1	6.0	7.0	8.0	9.0	12.2	16.3	22.8	100
5	0.0	0.0	0.0	6.6	9.4	11.6	12.6	13.6	15.2	17.8	21.9	28.4	100
6	0.0	0.0	86.5	91.3	95.5	98.0	100	100	100	100	100	100	100



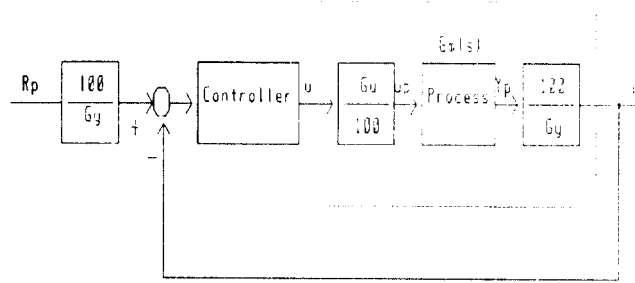


그림 3. 정규화된 제어기 시스템  
Fig. 3. Normalized Controller system

$T_{pn} = T_p / T_s$ 로 하여도 실제 제어 시스템과 같은 동작을 한다.

- 여기서,  $G_y$  : 프로세스 출력  $Y$ 의 정규화 이득  $\{Y_{pmax}\}$
- $G_u$  : 제어기 출력  $u$ 의 정규화 이득  $\{u_{pmax}\}$
- $e$  : 정규화된 제어오차
- $u$  : 정규화된 제어기 출력

2. 기준모델 선정

비례적분형 SOC에 의해 규정되는 1차계 기준모델의 선정은 2차원 성능지수  $PI(e^*, ce^*)$ 의 값이 프로세스가 정상상태로 도달할 때의  $e^*, ce^*$ 에 대해 식(20)과 같이 Zero 영역으로 수렴되어야 한다.

$$PI(e^*, ce^*) = (GE \cdot e)_q + (GCE \cdot ce)_q = e^* + ce^* = 0 \tag{20}$$

해석의 간편성을 위해 양자화를 무시하고 제어기는 연속시간으로 간주하면 식(20)은 식(21)로 나타낼 수 있다.

$$GE(R - Y) + GCE(\dot{R} - \dot{Y}) = 0 \tag{21}$$

여기서,  $e = R - Y$ ,  $R$ 은 상수로 간주하고  $\dot{R} = 0$ . 식(21)를 Laplace 변환하여 얻어진 1차계 기준모델은 식(22)와 같이 나타낼 수 있다.

$$Y = \frac{R}{1 + T_m S} \tag{22}$$

여기서,  $T_m = GCE/GE$  : 모델의 시정수.

제어기의 최적성능을 가지기 위해 비례적분형 SOC에 의해 규정되는 2차계 기준모델의 선정을 위해 요구되는 절충형 3차원 성능지수  $PI(e^*, ce^*, cce^*)$ 의 값은 2차원 성능지수  $PI(e^*, ce^*)$ 의 값에  $cce^*$ 의 값을 가산하여 취하고 프로세스의 정상상태에서는 식(23)과 같이 나타낼 수 있다.

$$PI(e^*, ce^*, cce^*) = PI(e^*, ce^*) + cce^* = e + ce + cce = 0 \tag{23}$$

식(23)을 정리하면 식(24)로 나타낸다.

$$GE \cdot R = GE \cdot Y + GCE \cdot \dot{Y} + GCCE \cdot \ddot{Y} \tag{24}$$

식(24)를 Laplace 변환하면 식(25)가 된다.

$$Y = R \cdot \frac{(GE/GCCE)}{(S^2 + (GCE/GCCE)S + GE/GCCE)} \tag{25}$$

여기서,  $\omega_n = (GE/GCCE)^{1/2}$  : 자연주파수  
 $\zeta = 1/2 \cdot (GCE^2 / (GE \cdot GCCE))^{1/2}$  : 제동계수.

한편 제어기의 사양으로서는 안정성에 관계되는 과도현상과 응답의 속용성으로 상승시간을 고려해야 하므로, 식(25)로부터 제어사양에 적합한 제어파라미터  $GE, GCE, GCCE$ 를 선택하여 기준모델을 선정할 수 있다.

V. 모의실험

모의실험에서 프로세스의 전달함수  $G_m(s)$ 는 식(26)를 이용하였다.

$$G_m(s) = \frac{0.262}{s^2 + 1.276s + 0.262} \quad (26)$$

식(26)에서  $T_p = 4.57$ (ms),  $T_s = 1.914$ (ms)인 프로세스의 정규화된 시정수  $T_{pn} = 2.39$ ,  $T_{sn} = 1$ , 샘플 수  $K$ 는 500으로 하였다. 제어 파라미터  $GE = 2.5$ ,  $GCE = 2$ ,  $GO = 1$ 로 고정하고,  $e^*$ 의 변화에 대한 GCCE를 표 5와 같이 가변시켰다.

표 5에서 보는 바와 같이  $|e^*|$ 가 적을 경우 GCCE를 크게하고  $|e^*|$ 가 클 경우 GCCE를 적게 취하도록 자동가변하여 제어기의 성능을 개선하였다. 반복과정과 시행착오를 거쳐 프로세스의 지연보상값(DEL)은 1, 2, 3으로 정하였으며 설정치는 50, 학습을 10회 하였을 경우 기존 SOC와 수정된 SOC와의 개선된 성능평가값을 표 6과 표 7에 나타내었다. 최적 학습수 7회를 기준으로 표6과 표7의 성능평가값 비교에서 수정된 SOC알고리즘이 우수함을 판단할 수 있다.

그림 4a, 4b에서 기존 SOC의 출력응답 특성은 과도 응답에 의해 나타나는 과,감쇠 진동을 연속변수를 가지는 제어알고리즘으로 제어파라미터를 자동으로 결정하여 과도응답을 개선시켰다. 정상상태에서 미세진동은  $e$ ,  $ce$ 에 대한 이득이 퍼지 추론부의 출력에 영향을 미치는 현상으로 나타나지만, 제안된 수정알고리즘에 의한 SOC에서는 경험적 발견지식에 의한 규칙수정 방법으로 정상상태에서 미세진동 현상을 효과적으로 감소시킬 수 있었다.

그림 5a, 5b, 5c는  $e$ ,  $ce$ 의 상관 관계에 의하여 정상상태에 대한 수렴정도를 나타내는 수정전, 후의 궤적도 비교로서 7회 학습시 가장 안정된 시스템으로 설정치에 도달함을 알 수 있으며, 수정 알고리즘에 의한 생성 규칙을 8a, 8b, 8c에 나타내었다. 모의실험의 결과에서 성능이 개선된 SOC는 수정전 SOC에 비해 과도응답, 목표 도달시간, 정상상태 오차에 있어 제어의 안정성이 향상되었음을 입증하였다.

표 5.  $e^*$ 에 대한 GCCE의 값  
Table 5. The value of GCCE for  $e^*$

$ e^* $	0	1	2	3	4	5	6
GCCE	2	2	1.7	1.4	1.1	0.8	0.5

표 6. 학습횟수 10번 시행시 기존 SOC의 성능평가 항목  
Table 6. Criteria for judging the performance on 10th learning number of a original SOC

학습횟수	IAE	OV	TS	TR
1	4.85	22.13	0.4433	0.3884
2	1.12	10.33	0.4387	0.425
3	1.11	4.98	0.4296	0.4204
4	1.19	4.86	0.4296	0.4204
5	1.2	6.22	0.4341	0.4204
6	1.09	5.97	0.4341	0.425
7	0.86	4.31	0.4296	0.4204
8	1.1	5.93	0.4341	0.4204
9	1.14	5.41	0.4341	0.4204
10	1.2	6.54	0.4387	0.4204

표 7. 학습횟수 10번 시행시 성능개선 SOC의 성능평가 항목  
Table 7. Criteria for judging the performance on 10th learning number of a improved SOC

학습횟수	IAE	OV	TS	TR
1	1.13	6.91	0.4433	0.4159
2	0.80	0.22	0.4387	0.4067
3	1.43	3.01	0.3884	0.3519
4	0.66	0.80	0.3839	0.3519
5	0.67	1.02	0.393	0.3427
6	0.67	1.02	0.393	0.3427
7	0.52	0.04	0.3793	0.3427
8	0.73	0.07	0.3884	0.3519
9	0.93	1.03	0.3839	0.3519
10	0.66	1.28	0.3884	0.3473

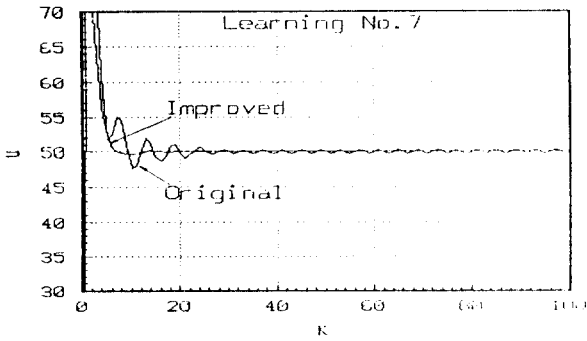


그림 4a. 학습횟수 7회시 제어기 출력응답 비교  
 Fig. 4a. Comparison with output response for the 7th learning number

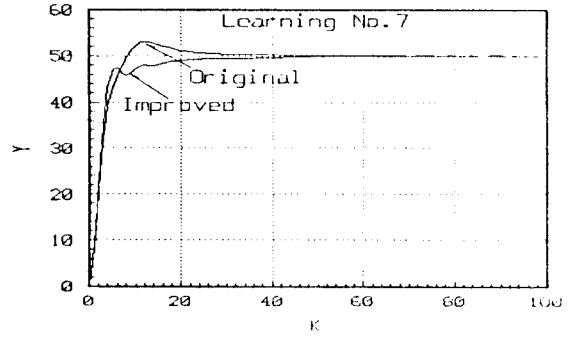


그림 4b. 학습횟수 7회시 프로세스 출력응답 비교  
 Fig. 4b. Comparison with process output response for the 7th learning number

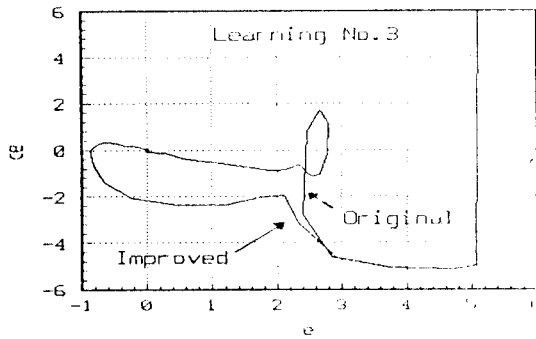


그림 5a. 학습횟수 3회시 e, ce 공간상의 궤적도 비교  
 Fig. 5a. Comparison with Trajectories diagram on (e, ce) space of the 3rd learning

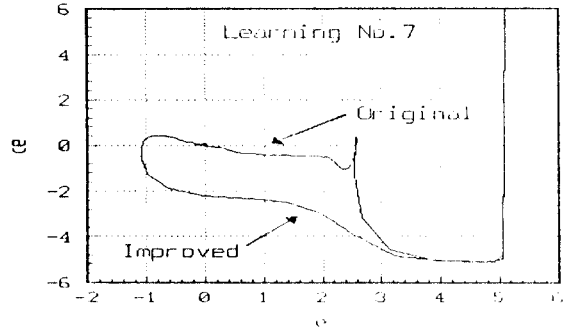


그림 5b. 학습횟수 7회시 e, ce 공간상의 궤적도 비교  
 Fig. 5b. Comparison with Trajectories diagram on (e, ce) space of the 7th learning

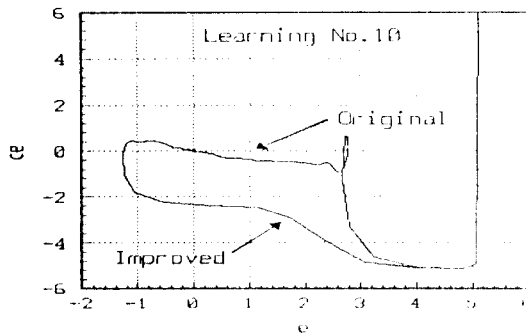


그림 5c. 학습횟수 10회시 e, ce 공간상의 궤적도 비교  
 Fig. 5c. Comparison with Trajectories diagram on (e, ce) space of the 10th learning

표 8a. 학습횟수 3번째의 생성규칙  
Table 8a. Rules produced by the 3rd learning

$e^* \ominus e^k$	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6
-6	*	*	*	*	*	*	*	*	*	*	*	*	*
-5	*	-22	-22	*	-22	*	*	*	*	*	*	*	*
-4	*	*	*	*	*	-9	*	*	*	7	5	*	*
-3	*	*	*	-8	-9	-9	*	-1	*	*	*	*	*
-2	*	*	*	-10	-7	-5	-2	0	2	*	*	*	*
-1	*	*	*	-8	-5	-2	-1	1	4	7	*	*	*
0	*	*	*	-9	-3	-1	0	0	2	8	*	*	*
1	*	-23	*	*	-5	-2	0	1	4	*	15	*	*
2	*	*	*	-4	-3	-1	2	5	6	*	*	*	*
3	*	*	-4	-3	-3	-1	0	0	5	7	*	*	*
4	*	*	-3	*	*	-1	0	5	5	*	*	*	*
5	*	-8	*	*	*	*	*	*	*	*	*	*	*
6	*	-9	*	*	*	*	*	*	*	*	*	*	96

표 8b. 학습횟수 7번째의 생성규칙  
Table 8b. Rules produced by the 7th learning

$e^* \ominus e^k$	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6
-6	*	*	*	*	*	*	*	*	*	*	*	*	*
-5	*	-22	-22	*	-22	*	*	*	*	*	*	*	*
-4	*	*	*	*	-10	-6	-6	*	*	7	5	*	*
-3	*	*	*	-10	-10	-6	-6	0	0	*	*	*	*
-2	*	*	*	-10	-7	-5	-2	1	1	*	*	*	*
-1	*	*	*	-8	-5	-2	-1	0	1	7	*	*	*
0	*	*	*	-9	-2	-1	0	0	1	8	*	*	*
1	*	-23	*	*	-2	-1	0	0	4	*	15	*	*
2	*	*	-6	-4	-2	-2	2	5	6	*	*	*	*
3	*	*	-7	-3	-3	-1	0	0	5	7	*	*	*
4	*	*	-3	*	*	-1	0	5	5	*	*	*	*
5	*	-8	*	*	*	*	*	*	*	*	*	*	*
6	*	-7	*	*	*	*	*	*	*	*	*	*	94

VI. 실험 및 결과 고찰

제안된 수정 SOC알고리즘에 의해 직류 서어보 모터의 속도제어를 하기 위한 실험장치를 그림 6에 나타내었으며, 실험장치의 전체 구조도는 그림 7에 도시하였다.

제어 알고리즘은  $\mu$ -콘트롤러인 80196을 이용하여 어셈블리 프로그램으로 실행하였다. 제작된 실험장치는 직류 서어보 모터인 TS198N56Z8 (100W용)의 엔코더로부터 출력되는 펄스를 분주시켜 펄스의 주기를 HSI-Data Available 인터럽트로 측정하여 속도를 검출하

표 8c 학습횟수 10번째의 생성규칙  
Table 8c. Rules produced by the 10th learning

$c \cdot \infty^*$	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6
-6	*	*	*	*	*	*	*	*	*	*	*	*	*
-5	*	-22	-22	*	-22	*	*	*	*	*	*	*	*
-4	*	*	*	*	-9	-9	-6	*	*	7	5	*	*
-3	*	*	*	-9	-9	-9	-6	2	1	*	*	*	*
-2	*	*	*	-9	-7	-5	-2	1	1	*	*	*	*
-1	*	*	*	-7	-5	-2	-1	1	1	7	*	*	*
0	*	*	*	-8	-3	-1	0	0	1	8	*	*	*
1	*	-23	*	*	-2	-2	0	1	4	*	15	*	*
2	*	*	-6	-4	-2	-2	2	5	6	*	*	*	*
3	*	*	-7	-3	-3	-1	0	0	5	7	*	*	*
4	*	*	-6	*	*	-1	0	5	5	*	*	*	*
5	*	-8	*	*	*	*	*	*	*	*	*	*	*
6	*	-7	*	*	*	*	*	*	*	*	*	*	90

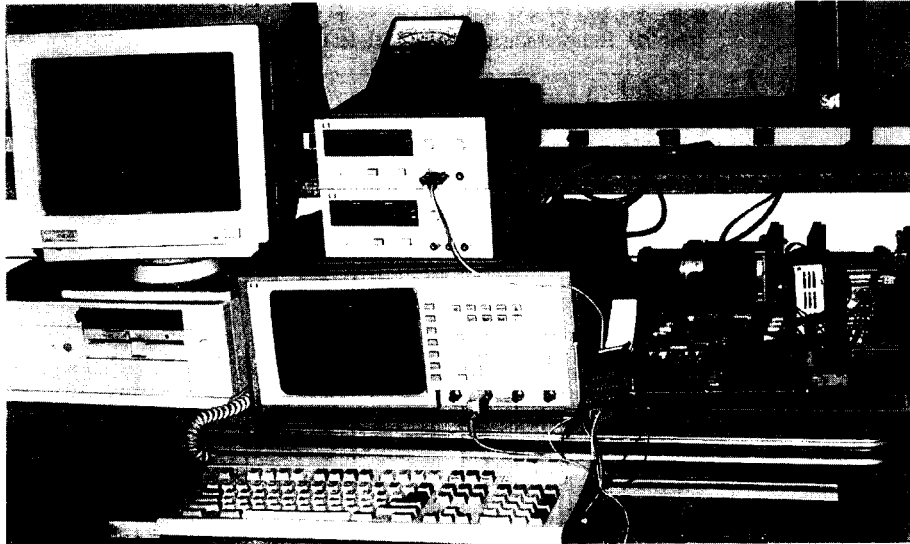


그림 6. 실험장치  
Fig. 6. Experimental apparatus

었다.

제어기의 출력은 80196의 PWM 출력단자를 이용하였으며, 스위칭 전력소자는 IGBT인 1MBH60-100을 사용하고, 각부의 제어신호는 PAL인 22V10 IC로서

논리설계하였다. 모든 데이터는 RAM을 이용하고 ROM에서 80196 어셈블리어로 주 프로그램을 ROM Write하였으며, 제어기의 출력신호와 속도감출신호는 D/A인 1408로서 변환시켜 디지털 오실로스코프로 관

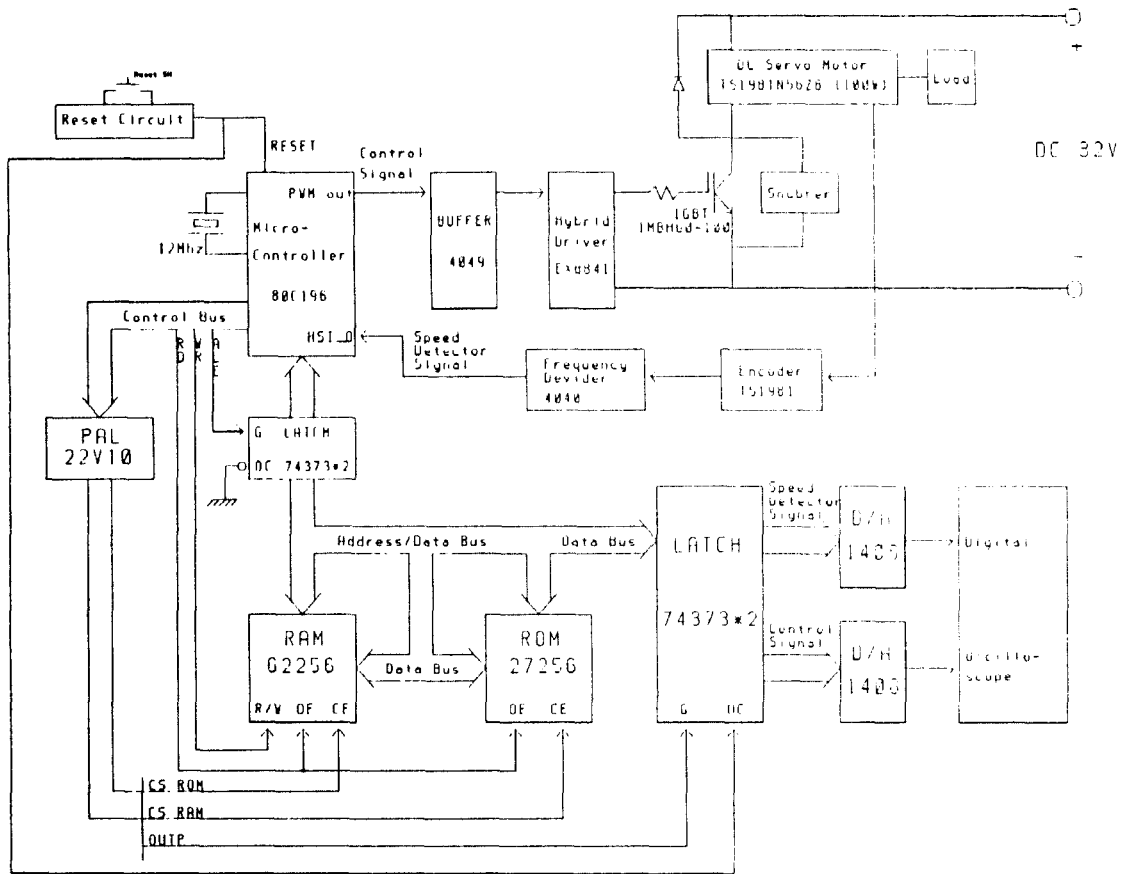


그림 7. 실험장치 구조도  
Fig. 7. Experimental equipment configuration

측하였다. 실제 모터의 사양은 표 9에, 이것의 전달함수는 식(26)을 이용하였다.

수정 알고리즘에 대한 실 시간 제어의 성능을 측정기 위해 설정치는 4[V]의 구형응답으로 하고, 샘플링 주기  $t_s$ 는 4.57[ms], 지연보상 DEL의 값은 모의실험의 결과에 따라 1, 2, 3으로 하였을 때 무부하 응답특성에 대한 제어기 출력  $u$ , 프로세스 출력  $Y$ 를 그림 8에 나타내었다.

모의실험 결과에서 응답특성의 최적 학습수는 7번째 임을 확인하였고, 7회까지의 학습 완료상태에서 생성된 규칙으로 직류 서어보 모터를 구동시켰다. 제어 파라미터의 값은 시스템을 안정화 시키기 위하여 GE는 0.3, 0.5, 0.8,로 하고 GCE는 2, 가변, 그리고 GCCE는

표 9. 모터의 사양  
Table 9. Specification of the motor

모델명	TS198N56Z8
Torque Constant( $K_T$ )	0.828 kg-cm/A (11.5 OZ-IN/A)
Voltage Constant( $K_E$ )	8.5 <sup>10%</sup> V/K rpm
Armature Resistance( $R_a$ )	0.92 $\Omega$
Armature Inductance( $L_a$ )	0.9 mH
Max. Speed(N)	4000 rpm
Moment of Inertia( $J_M$ )	0.285 g-cm-sec <sup>2</sup> (3.85 10 <sup>-3</sup> OZ-IN-sec <sup>2</sup> )
Mechanical Time Constant( $\tau_m$ )	3.9 msec
Electrical Time Constant( $\tau_e$ )	0.98 msec

0.5, 0.8, 가변의 방법으로 출력응답 특성을 비교 고찰하였다. 이와같이 제어 파라메터 값의 조건에 따라 출력 특성의 변화를 보이지만, 반복된 시행착오의 실험을 한 결과 GE는 0.8, GCE는 가변, 그리고 GCCE는 가변했을 때 최적의 출력응답 상태가 됨을 알 수 있었다. 또한 프로세서의 출력 Y 에서는 1회 학습에서 충분한 제어 성능을 발휘치 못하고 과도 현상을 일으키나, 2회에서 7회 까지의 학습과 학습후 생성된 규칙으로 시어보 모터에 접속하였을때 제어계가 안정함을 나타내었다.

그림 9는 부하시의 출력응답 특성으로 직류 서어보 모터의 축에 발전기 부하로써 직류 서어보 모터인 TS3143N1E263(TAMAGAWA SEIKI사 제품 150W)을 결합하여 제어 파라메터 GE, GCE, 그리고 GCCE의 값을 가변하였을때의 응답 특성을 비교하였다. 무부하시와 같은 조건으로 하여 GE는 0.8, GCE는 자동가변, 그리고 GCCE 역시 자동 가변으로 하여 학습(1회 - 7회)후 수정, 생성된 규칙으로 8회 이후 부

하를 출력에 인가했을 경우 부하시의 출력 응답은 무부하시에 비해 초기 과도 응답은 불안정 함을 나타내 보이지만 전반적인 특성은 안정되게 제어됨을 확인할 수 있었다.

### VII. 결 론

본 연구에서 얻은 결론은 다음과 같다.

1. 제안된 수정 SOC알고리즘에 의해 직류모터의 속도제어의 실시간 처리를 위해  $\mu$ -콘트롤러인 80196을 이용하여 제어시스템을 실현시켰다.
2. 제어오차의 감소를 위해 입, 출력 변수를 비선형 양자화 시키고 가중치의 도입으로서 연속 변수화 하였으며 정상상태에서의 제어오차의 감소를 위해 heuristic knowledge에 의한 규칙수정을 행하였다.
3. 제어규칙과 성능지수표는 2차원 비례적분 요소에 cce\*의 함을 절충하여 3차원 형태의 성능지수값을 생성

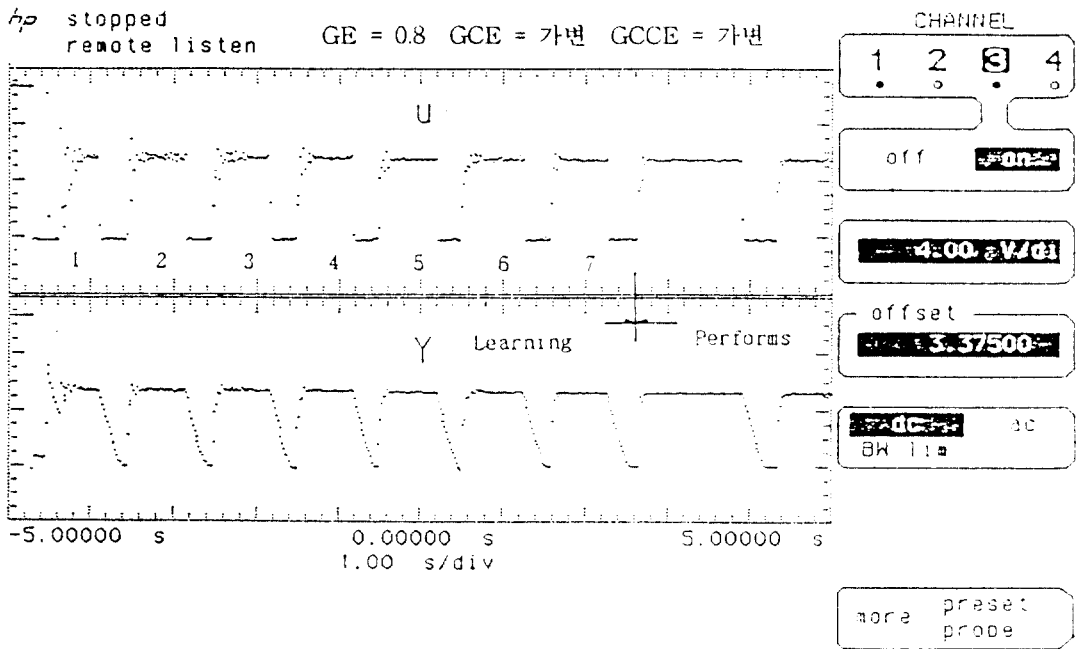


그림 8. 무부하시의 제어시스템의 제어기 출력 u, 프로 세스 출력 Y  
 Fig. 8. Controller output u and process output Y of the control system with no load

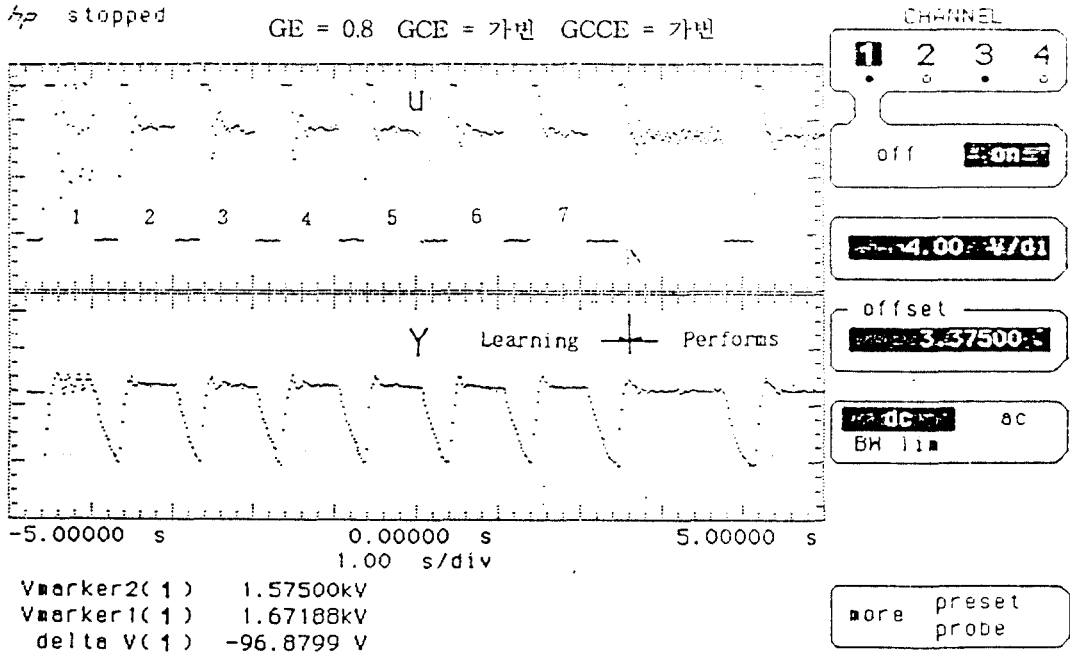


그림 9. 부하시의 제어시스템의 제어기 출력 u, 프로세스 출력 Y  
 Fig. 9. Controller output u and process output Y of the control system with load

시켜 수렴특성을 개선 하였으며,

4. 프로세스의 특성이 기준모델에 적용되도록 제어 파라미터를 가변하였고, 입,출력 변수 및 성능지수표의 비선형 양자화를 행하여 동 특성범위의 확대와 응답의 속응성을 기하였다. 프로세스와 제어기에 대해  $G_u$ 와  $G_y$ 의 이득을 도입하여 제어시스템의 정규화로서 변수의 동적 범위가 실시간 제어에 적용되도록 하였다.

본 연구에서 알 수 있듯이 실시간 능력을 갖는 하드웨어 구현의 효용성은 입증되었으나, 학습 완료후 부하 인가시 초기 과도응답의 최소화 방법과 학습 회수에 의존치 않는 적합도 함수에 대한 지식기반 실시간 제어기의 연구가 요구된다.

참고문헌

1. L.A. Zadeh, "Fuzzy sets", Inform. Contr., Vol.

8, pp.338-353, 1965.  
 2. Y.F.Li and C.C.Lau, "Development of Fuzzy Algorithms for Servo systems", IEEE Control Systems Magazine, pp.65-71, April, 1989.  
 3. T.J. procyk, "A Linguistic Self-organizing Process Controller", Automatica, Vol.15, No.1, pp.15-30, 1979.  
 4. W.C. Daugherty and J.Yen, "Performance Evaluation of a Self-Tuning Fuzzy Controller", IEEE Int'l conference on Fuzzy Systems, Vol.1, pp.389-397, March, 1992.  
 5. Mikio Maeda, "Self-tunning Fuzzy Controller", 計測自動制御學會, Vol. 24, NO.2, pp.191-197, 1988.  
 6. R.Katz and H.Boyet, "The 16 bit 8096 programming, Interfacing and Applications", MTI,



1986.

7. T. Yamazaki, "An Improved Algorithm for a Self-organizing Controller", Ph.D Thesis, Queen Mary college, Univ. of London, 1982.

8. L. Zheng, "A Practical Guide to Tune of proportional and Integral (PI) Like Fuzzy Controllers", IEEE Int'l conference on Fuzzy Systems, Vol.1, pp.633-640, March, 1992.



姜 信 出 (Shin Chul Kang) 정회원

1955년 10월 13일생  
 1988년 2월 : 부산공업대학 전기공학  
 과(공학사)  
 1990년 2월 : 동아대학교 전자공학  
 과(공학석사)  
 1995년 2월 : 동아대학교 전자공학  
 과(공학박사)

1991년 4월~1992년 5월 : 한국통신보은위성지구국 과장  
 1992년 5월~현재 : 한국통신 부산해저중계국 과장  
 \*주관심 분야 : 마이크로프로세서, 퍼지제어



林 永 道 (Young Do Lim) 정회원

1949년 9월 20일생  
 1973년 2월 : 동아대학교 전자공학  
 과(공학사)  
 1978년 2월 : 동아대학교 전자공학  
 과(공학석사)  
 1987년 8월 : 동아대학교 전자공학  
 과(공학박사)

1988년 3월~현재 : 동아대학교 전자공학과 부교수  
 \*주관심 분야 : 신경망제어, 퍼지제어, 전력전자, 고주파 유  
 도가열