

Expert형 제어 기법에 의한 자기 동조 제어기에 관한 연구

A Study on the STC Utilizing Expert Control Technique

蔡 昌 鉉* · 李 昌 勳** · 任 殷 彬[§] · 禹 廣 芳^{§§}

(Chang-Hyun Chai · Chang-Hoon Lee · Eunbin Yim · Kwang-Bang Woo)

요 약

디지털 STC의 체계적 동조에 있어서 미지이며 시변인 매개변수 그리고 지연시간을 가진 시스템인 경우, 다수의 전제 조건과 문제점 때문에 현재까지는 개개의 경우를 별도로 해결하려는 연구만 진행되어 왔다. 따라서 복잡화된 제어 시스템의 문제점을 효율적으로 해결하기 위한 Expert형 제어 기법이 절실히 요구되고 있다. 본 연구에서는 공정제어를 위해 Expert형 제어기, 즉 Expert형 제어기법에 의한 자기 동조 제어기를 제안한다. 이 Expert형 제어기는 제어 대상의 현재 상태를 감시하여 불안정한 상태를 지시하는 불안정 상태지시기, 이 불안정 상태지시기에 의해 동작 모드가 변경되는 Expert형 Back-up 제어기 및 Expert형 STC, 그리고 지식기반 및 Inference Engine으로 이루어지는 전문가 시스템으로 구성된다. 제안된 Expert형 제어기는 1) 계통의 지연시간이 미지이고 시변인 시스템, 2) 계통의 매개변수가 미지이고 시변인 시스템, 그리고 3) 최소 위상뿐만 아니라 비최소 위상 시스템등에 대해 제어기능을 성공적으로 수행 가능하다. 이 견실한 제어기능은 컴퓨터 시뮬레이션에 의해 입증되었다.

Abstract- In systematic tuning of digital STC parameters, systems with time-varying parameters and systems with time-varying delays are studied individually because of many preconditions and difficulties. In order to eliminate all these difficulties effectively, the expert control technique is required to enhance STC control functions. In this paper, an expert controller, a STC utilizing expert control technique, for process control is designed. The expert controller is composed of an instability indicator for detecting plant instability, an expert back-up controller and an expert STC which are switched each other by the instability indicator, and expert system with knowledge base and inference engine. This expert controller is able to perform control functions successfully for the following; 1) a system which has unknown and time-varying delay time, 2) a time-varying system which has unknown parameters, and 3) a system with minimum and non-minimum phase. The robust control function is demonstrated by computer simulations.

1. 서 론

최근 공정 제어 분야에서 보다 향상된 제어 수행을 위하여 다양한 제어기가 개발되고 있으나, 실제 공정의 다양하고 복합적인 문제점 극복을 위하여 숙련된 전문가의 경험적 지식이 절실히 요구되고 있다. 그러나 이와같은 경험적 지식(Heuristic Knowledge)을 기존의 언어로 수식화 하는 데 많은 어려움이 따른다. 따라서, 이러한 문제점의 해결을 위하여 제어 시스템에 AI기법을 도입하는 전문가 제어가 활발히 진행되고 있다.

제어 시스템에 AI기법을 도입하는 방법에는, 첫째 기존의 제어 알고리즘에 의한 출력을 감시하여 문제점 발생시 경험적 지식에 의해 처방을 하는 관리 제어(Supervising Control)[11-23]와, 둘째 제어 알고리즘 자체에 AI기법을 도입하는 지능형 제어기(Intelligent Controller)[24, 25]를 설계하는 방법이 있다. 전자의 방법으로 Astrom등[12]은 단일 페루프 시스템에 적용하여 제어기가 수동, 동조, 고전 PID 모드가 선정되도록 약 70여개의 규칙을 적용하였다. 이 경우에는 대형 컴퓨터에 의해 시행되어 PC-Level에서 적용하기에는 문제점이 있다. Jiang등[20, 21]은 전문가 시스템을 계층적 구조로 구성하여 모든 의사 결정과 오점 탐지(Fault Detection)를 관리 구조에서 수행하고, 제어 구조로는 PID 제어를 사용하여 실시간에 적용하였으나, 페루프 극점을 미리 알아야 하며, 페루프 극점이 복소 쌍대값을 가져야 감시할 수 있다는 단점이 있다. 후자에 의한 방법으로 Sripada등[24, 25]은 AI기법과 Fuzzy 논리방법을 결합하여 프로세스의 실시간 제어에 적용하였으나, 이 AI 제어기는 Servo 제어기와 조절 제어기로 분리되며, 대상 프로세스에 따라 Fuzzy 논리를 위한 Fuzzy Set Membership 함수를 정의하여야 하고, 규칙 적용마다 정의된 변수의 결정에 따른 불확실성(Uncertainty)이 발생하는 단점이 있다.

본 연구에서는 공정 제어에 적합한 Expert형 제어기, 즉 Expert형 제어 기법에 의한 자기 동조 제어기를 구성하기 위하여 플레트의 현재 상태를 감시하여 불안정한 상태를 지시하는 불안정 상태

지시기[3, 10]를 개발하고, 지시기에 의해 동작 모드를 변경할 수 있도록 Expert형 Back-up 제어기 및 Expert형 STC의 Scheduler를 개발한다.

따라서 본 연구에서 구성되는 Expert형 제어기는 1) 계통의 지연 시간이 미지이고 시변인 시스템, 2) 계통의 매개 변수가 미지이고 시변인 시스템, 그리고 3) 최소 위상뿐만 아니라 비최소 위상 시스템 등에 적용하여 문제점 발생시에 제어기가 스스로 감시하여 처방하도록 구성된다.

2. Expert형 제어기의 구조

Expert형 제어기의 구조는 전문가 시스템, 수치 알고리즘, 사용자 인터페이스의 세 부분으로 나누어지며 전체 시스템 구조는 그림1과 같다.

전문가 시스템은 지식기반과 Inference Engine으로 구성된다. Inference Engine은 다시 Scheduler와 Interpreter로 나뉘는데, 본 연구에서는 Scheduler의 일부와 Interpreter는 Turbo-Prolog에서 제공되는 기능을 이용하였다. 지식기반은 다양한 형태의 규칙으로 이루어지며 모든 규칙들은 If (Situation), Then (Action)의 형식으로 구성되어 (Situation)은 제어대상의 상태를 기술하고 (Action)은 제어 입력의 적절한 변화를 지시한다. Expert형 제어기를 구성하는 중요한 규칙들은 다음과 같다.

- (1) Expert형 Back-up 제어기의 서어보제어 입력을 조절하는 규칙군,
- (2) Expert형 Back-up 제어기의 절환요소 e^* 를 조절하는 규칙군,
- (3) Expert형 Back-up 제어기의 조절제어 활성화 규칙군,
- (4) Expert형 Back-up 제어기의 조절제어 제한값 규칙군,
- (5) Expert형 STC의 활성화 규칙군,
- (6) Expert형 STC의 제한값 규칙군,
- (7) 불안정 상태지시기의 규칙군,

이와 같은 지식기반은 수치형 알고리즘에 근거를 두고 있으며 Scheduler에 의해 Expert형 STC와 Expert형 Back-Up 제어기의 동작 모드가 결정된다. 본 연구에서 사용될 지식기반과 Scheduler는 제3장에 상술한다.

수치형 알고리즘은 제어 알고리즘과 동정 알고리즘으로 나뉜다. 제어 알고리즘은 주 제어기로 Cameron과 Seborg[1]가 제안한 적용 PID 제어 알고리즘을 사용하고, 보조 제어기로 Sripada가

*正 會 員 : 金島工大 電子制御工學科 專講・工博

**正 會 員 : 延世大 大學院 電氣工學科 博士過程

§正 會 員 : 延世大 大學院 電氣工學科 碩士過程

§§正 會 員 : 延世大 工大 電氣工學科 教授・工博

接受日 字 : 1988年 9月 27日

1次修正 : 1989年 5月 22日

2次修正 : 1989年 6月 14日

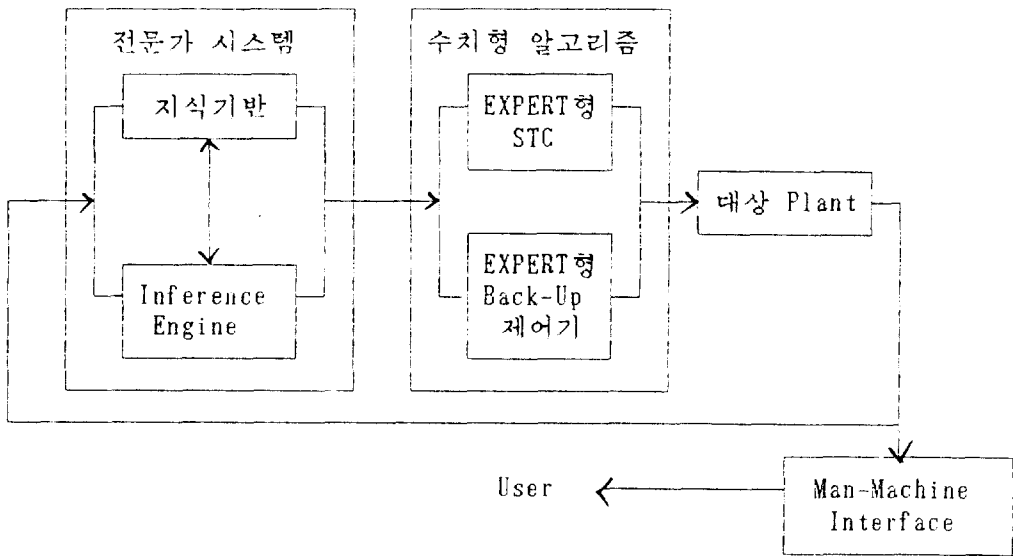


그림 1 Expert형 제어기의 구조도
 Fig. 1 The Structure of Expert Controller

제안한 지능형 제어기를 개선하여 사용한다. 동정 알고리즘은 확장된 최소 자승 알고리즘이다.

Expert형 제어기를 효율적으로 운영하기 위한 인터페이스는 Turbo-C와 Turbo-Prolog간의 인터페이스, 그리고 Turbo-Prolog와 사용자와의 인터페이스로 이루어진다.

Turbo-Prolog로는 지식 기반과 Scheduler를 구성하고, Turbo-C로는 기준입력 발생 모듈, 미지의 매개 변수를 추정하기 위한 동정 모듈, 시스템 시뮬레이션 모듈, 각 변수들의 초기값을 주기 위한 초기화 모듈, Scheduler에 필요한 값들을 읽어들이는 모듈, Scheduler에서 발생된 값을 저장하는 모듈 등이 포함되어 있다. Prolog와 C로 구성된 이러한 각 모듈들은 Turbo-C와 Turbo-Prolog간의 인터페이스를 위하여 cpinit이라는 global predicate로 선언하면, Prolog에서의 predicate와 마찬가지로 C모듈들이 취급된다. 이 결과 인공 지능용 언어인 Prolog와 제어언어인 C와의 인터페이스가 이루어진다.

또한 사용자와의 친밀도를 높이기 위하여 Turbo-Prolog에서 제공되는 window 기능을 이용하여 공정 제어 상태 표시 window, Dialog window, 시스템의 출력이 표시되는 출력 window 등으로 화면을 구성한다.

3. 지식 기반 및 제어용 Scheduler

3.1 Expert형 Back-up 제어기의 지식 기반 및 Scheduler

Expert형 Back-Up 제어기는 Rule of Thumb에 의한 제어 기법으로 2 level Bang-Bang 제어 이론에 그 근거를 두고 있다. 주 제어기인 STC 알고리즘에 의한 제어가 불가능할 때, 시스템 출력에 따라 구축된 지식 기반에 의해 제어 입력을 계산해 주는 제어기로서, 출력을 가능한 빨리 설정점에 도달하도록 하는 서어보 제어기와 설정점 근처에서 매끄러운 출력을 얻도록 하는 조절 제어기로 구성된다.

1) 서어보 제어를 위한 지식 기반

서어보 제어용 지식 기반은 크게 제어입력을 조절하는 규칙군과 절환 요소 e^* 를 조절하는 규칙군으로 구분된다. 제어 입력을 조절하는 규칙군은 세어오차 $e(k)$ 와 절환요소 e^* 의 상대적 크기에 따라 다음과 같이 결정된다.

- (1) 기준입력이 양의 방향으로 변화시
 - i) If $e(k) < e^*$, then $u(k) = U_0 + e(k)/ed$
 - ii) If $e(k) > e^*$, then $u(k) = U_{min}$

구간 N Sampling 시간내
- (2) 기준입력이 음의 방향으로 변화시
 - i) If $e(k) < e^*$, then $u(k) = U_0 - e(k)/ed$
 - ii) If $e(k) > e^*$, then $u(k) = -U_{min}$

구간 N Sampling 시간내

여기서 U_0 는 제어입력의 초기값으로 0 이고

U_{min} 는 제어입력의 제한값으로 $0.5 u(k)$ 이며 ed 는 Expert형 STC에서 결정되는 절분값으로 양의 상수이다.

이 규칙군은 기준입력의 변화시에만 순차적으로 작동되나, 또한 절환 요소를 조절하는 규칙군은 기준 입력이 양의 변화시에 선정되면, 음의 변화시에는 작동되지 않는 규칙군으로 다음과 같다.

(1) 출력 $y(k)$ 가 기준 입력보다 크면 Overshoot이므로 $e^* = e^* - \Delta e$

(2) 출력 $y(k)$ 가 기준 입력보다 작으면 Overdamping이므로, $e^* = e^* + \Delta e$

여기서 Δe 는 설계자가 정하는 값으로 기준입력의 1/20 정도로 한다.

2) 조절 제어를 위한 지식 기반

본 연구에서 제어 입력을 결정하기 위하여 조절 제어용 지식 기반을 다음과 같이 2개의 규칙군으로 분류한다.

(1) 활성 규칙군 : 매 제어 구간에서 하나 이상 항상 사용되는 규칙군

(2) 제한값 규칙군 : 프로세스의 출력이 제한값 근처에 있을 때에만 적용되는 규칙군

이중 활성 규칙은 상황 묘사를 규칙화한 것으로 다음과 같이 분류되고 각 규칙에 대한 부 규칙은 그림2와 같다.

- (1) 오차에 관한 규칙
- (2) 경사도에 관한 규칙
- (3) 제어 행위 증감에 관한 규칙

그림2에서와 같이 기준입력의 변화시 부호에 따

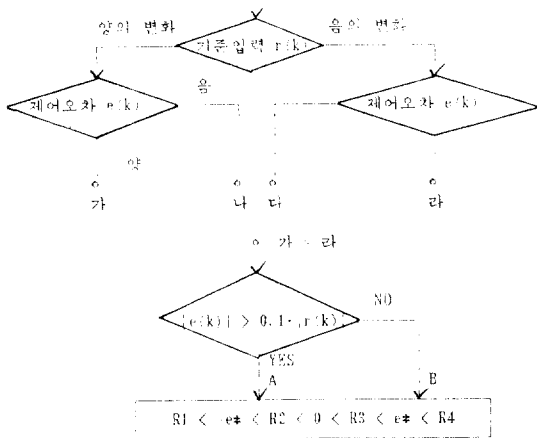


그림 2 활성규칙군의 세부규칙
Fig. 2 The Subrules in the Group of Active Rule

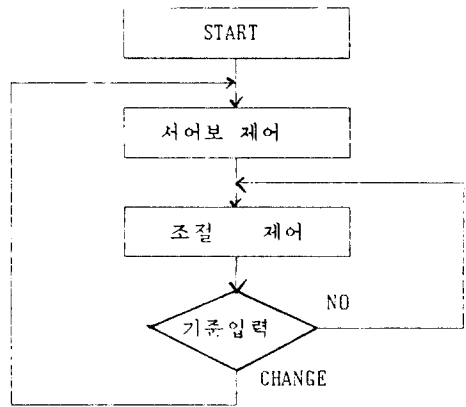


그림 3 Expert형 Back-up제어의 Scheduler
Fig. 3 The Scheduler of Expert Back-up Controller.

라 활성 규칙군은 반으로 감소되고, 제어 오차의 부호에 따라 활성 규칙군이 반으로 감소되며, 제어 오차의 범위에 따라 활성 규칙군이 반으로 감소된다. 여기에서 $R1, R2, R3, R4$ 는 그림2의 가~라에 모두 같이 적용되는 제어입력에 대한 증감의 양으로 $|R1|=|R4|, |R2|=|R3|$ 이다. 따라서 AI 제어기는 Breadth-first 기법으로 최소 4번, 최대 7번의 Tracking으로 제어 입력의 변화량을 계산하여 규칙 적용에 따른 결정시간을 단축한다.

프로세스 출력이 제한값 근처에 있을 때에 적용되는 규칙은 다음 3 종류로 한 규칙은 제한 영역에서 더해지는 부가적인 제어 행위의 양을 명시한다. 나머지 2개의 규칙군은 프로세스 출력이 제한 영역에 도달했을 때, 제한 영역을 떠날 때, 그리고 제한 영역에 있을 때 발생하는 전체 행위에 대한 것으로 다음과 같다.

- (1) 출력 $y(k)$ 가 제한 영역에 있을 때, 제어 입력의 큰 변화
- (2) 출력 $y(k)$ 가 제한 영역으로 들어올 때, 제어 입력의 큰 변화를 누진
- (3) 출력 $y(k)$ 가 제한 영역에서 나갈 때, 누진된 제어 입력의 변화를 감산

위에서 설명한 지식기반에 의한 Expert형 Back-up 제어기의 Scheduler는 그림3과 같다.

3.2 Expert형 STC의 지식 기반 및 Scheduler

식(1)과 같은 단일 입출력 모델에 대해 Cameron과 Seborg[1]가 제안한 STC를 구성한다.

$$A(z^{-1})y(t) = z^{-k-1}B(z^{-1})u(t) + C(z^{-1})\xi(t) \quad (1)$$

여기서 $y(t)$ 는 측정 출력, $u(t)$ 는 제어 입력, $\xi(t)$ 는 백색 잡음, t 는 샘플링 시간이다.

다항식 $A(z^{-1}), B(z^{-1})$ 와 $C(z^{-1})$ 는 후향 변위 연산자 z^{-1} 의 항으로 표시되는 식(2)의 다항식이다.

$$\begin{aligned} A(z^{-1}) &= 1 + a_1z^{-1} + \dots + a_nz^{-n} \\ B(z^{-1}) &= b_0 + b_1z^{-1} + \dots + b_mz^{-m}, b_0 \neq 0 \\ C(z^{-1}) &= 1 + c_1z^{-1} + \dots + c_nz^{-n} \end{aligned} \quad (2)$$

여기서 $C(z^{-1})$ 의 모든 근들은 단위원 내에 존재한다고 가정한다.

STC는 식(3)의 보조출력 $\phi(t)$ 의 변위를 최소화시키도록 설계한다.

$$\phi(t) = P(z^{-1})y(t) + Q(z^{-1})u(t-k-1) - R(z^{-1})y_r(t-k-1) \quad (3)$$

여기서 $y_r(t)$ 은 기준 입력, 그리고 P, Q 와 R 은 제어기 매개변수를 결정하는 전달함수로 식(4)와 같다.

$$P(z^{-1}) = P_n(z^{-1})/P_a(z^{-1}), Q(z^{-1}) = Q_n(z^{-1})/Q_d(z^{-1}), R(z^{-1}) = R_n(z^{-1})/R_d(z^{-1}) \quad (4)$$

시간 $t-k-1$ 에서 $Q \cdot u(t-k-1)$ 은 알고 있으므로, 보조 출력 $\phi(t)$ 를 예상하는 문제는 $(t-k-1)$ 까지의 입출력 데이터가 주어졌을 때 식(5)를 예상하는 문제이다.

$$\phi_y(t) = P \cdot y(t) \quad (5)$$

따라서

$$(a) \text{ 예상식: } C\phi_y^*(t/t-k-1) = (F/P_a) \cdot y(t-k-1) + G \cdot u(t-k-1) \quad (6)$$

$$(b) \text{ 예상오차: } \phi_y(t) = \phi_y(t/t-k-1) + E\xi(t) \quad (7)$$

여기서 다항식 $E(z^{-1})$ 와 $F(z^{-1})$ 는 식(8)로 주어지는 다항식이고 $G = EB$ 이다.

$$(CP_n/AP_a) = E(z^{-1}) + z^{-k-1}(F/AP_a) \quad (8)$$

따라서 식(1)과 식(8)을 식(5)에 대입하여 식

(9)의 예상모델을 얻을 수 있다.

$$\begin{aligned} \phi(t+k+1) &= (F/CP_a) \cdot y(t) + (EB/C+Q) \cdot u(t) \\ &- R \cdot y_r(t) + E\xi(t+k+1) \\ &= \phi^*(t+k+1/t) + \tilde{\phi}(t+k+1/t) \end{aligned} \quad (9)$$

$$\begin{aligned} \text{여기서 } \phi^*(t+k+1/t) &= (F/CP_a) \cdot y(t) + (EB/C+Q) \cdot u(t) - R y_r(t) \\ \tilde{\phi}(t+k+1/t) &= E\xi(t+k+1) \end{aligned}$$

식(9)에서 ϕ^* 는 $(k+1)$ -step 앞선 예상식이고 $\tilde{\phi}$ 는 예상 오차이다. 따라서 예상오차 $\tilde{\phi}(t+k+1/t)$ 를 0이 되게 하고, $\phi(t)$ 의 변위를 최소로 하기 위하여 $\phi(t+k+1)=0$ 인 조건에서 제어 법칙의 식(10)을 얻을 수 있다.

$$u(t) = [CR \cdot y_r(t)/P_a] / (EB+CQ) \quad (10)$$

만약, 시스템의 매개변수 A, B 와 C 를 안다면, E 와 F 는 식(8)에서 구하여 식(10)을 만족하는 제어 입력을 계산한다. 시스템의 매개변수를 모르는 경우, 이 제어법칙은 식(9)의 예상 모델과 더불어 STC를 위한 기초를 제공해 준다.

본 연구에서 이용된 예상 모델을 식(5)와 같이 출력 $y(t)$ 에만 종속적인 $\phi(t)$ 의 성분, $\phi_y(t)$ 를 기초로 하므로 식(9)는 식(11)과 같이 변형된다.

$$\phi_y(t+k+1) = (F/CP_a) \cdot y(t) + (EB/C) \cdot u(t) + E\xi(t+k+1) \quad (11)$$

만약 $C=1$ 로 가정하면, 식(11)은 식(12)와 같으므로 선형 최소 자승 추정기법을 이용하여 시스템의 매개변수를 추정할 수 있다. 본 연구에서는 RELS기법을 적용하여 최대 지연시간이 $k=t/t=4$ 까지 변화해도 견실한 매개변수 추정이 되도록 한다.

$$\phi_y(t+k+1) = F \cdot y_r(t) + G \cdot u(t) + \varepsilon(t+k+1) \quad (12)$$

$$\begin{aligned} \text{여기서 } y_r(t) &= 1/CP_a y(t) \\ \varepsilon(t+k+1) &= E\xi(t+k+1) \text{이다.} \end{aligned}$$

식(12)에서 F 와 G 는 z^{-1} 로 표시되는 다항식이고, $\varepsilon(t+k+1)$ 는 잡음 $\xi(t)$ 의 이동평균이다.

식(10)을 다시 쓰면

$$u(t) = [R \cdot y_r(t) - F \cdot y_r(t)] / (G+Q) \quad (12)$$

식(13)이 PID구조를 갖기 위해서 F다항식의 차수가 2가 되도록 선정한다. 또한 출력 $y(t)$ 와 기준 입력 $y_r(t)$ 가 정상상태에서 일치하도록 R을 선정하면 식(14)와 같다.

$$R = [F/P] = (\hat{f}_0 + \hat{f}_1 z^{-1} + \hat{f}_2 z^{-2}) / (1 + P1Z^{-1}) \quad (14)$$

여기에서 \hat{f}_i 는 다항식 F의 계수, P1은 다항식 P의 일차계수이다.

최종 단계로 기준입력만 적분항에 포함하기 위하여 다음과 같이 선정한다.

$$G + Q = (1 - z^{-1}) / \nu \quad (15)$$

식(15)에서 ν 는 설계변수로 0~1의 값이다. 이 방법에 의하면 기준 입력이나 부하의 변동이 있을 후에도 잔류 편차를 제거할 수 있다. 식(14), 식(15)를 식(13)에 대입하여, 기준입력만 적분항에 포함하는 PID 구조를 갖는 STC를 얻는다.

$$\nabla u(t) = \nu [R \cdot y_r(t) - (\hat{f}_0 + \hat{f}_1 z^{-1} + \hat{f}_2 z^{-2}) y_r(t)] \quad (16)$$

여기서 출력 필터는 다음과 같은 형태를 갖는다.

$$y_r(t) = y(t) / (1 + P1z^{-1}), -1 < P1 \leq 0 \quad (17)$$

따라서, 기준 입력이 적분항에만 포함하는 PID 제어기의 식(18)과 비교하면 제어기 매개변수들 식(19)와 같이 유도할 수 있다.

$$\begin{aligned} \nabla u(t) &= K_c \cdot [-y_r(t) + y_r(t-1) + (T_s / T_i) \cdot (y_r(t) - y_r(t-1)) + (T_d / T_s) \cdot (-y_r(t) + 2y_r(t-1) - y_r(t-2))] \\ &= (K_s T_s / T_i) \cdot y_r(t) - K_c [1 + (T_s / T_i) + (T_d / T_s)] \cdot y_r(t) + \\ &K_c \cdot [1 + (2T_d / T_s)] \cdot y_r(t-1) - (K_c T_d / T_s) \cdot y_r(t-2) \end{aligned} \quad (18)$$

$$\begin{aligned} \text{여기서 } K_c &= -\nu(2\hat{f}_2 + \hat{f}_1) / \alpha \\ T_i &= (T_s K_c) / \{[\nu \hat{f}_0 / \alpha] - K_c(1 + T_d / T_s)\} \\ T_d &= T_s \cdot \nu \cdot \hat{f}_2 / (\alpha - K_c) \\ &= 1 + P1 \end{aligned} \quad (19)$$

$$(20)$$

STC의 설계변수 ν 는 제어이득 K_c 와 똑같은 효과를 갖는다. 즉, ν 값이 크면 결과적으로 과다한 제어가 되어 부족감쇄 응답이 나타나고, 반면에 ν 값이 작으면 보다 완만한 제어로 과다감쇄 응답이

나타난다. $\nu=0$ 일때 개루프 제어가 된다.

Cameron과 Serborg는 식(15)를 만족시키기 위하여 Clarke-Gawthrop의 접근법을 개선하여 설계 변수 값을 연역적으로 추정하는 접근법을 사용하였다. 그러나 이 방법은 제어대상의 안정도를 보장하기 위하여 최대설계변수 값을 수정 Routh-Hurtz 판별법이나 Jury 판별법에서 구해야 하며 지연시간의 변화에 따라 최대설계변수 값을 계산 정해야 하므로 최적지연시간을 추정해야 하는 단점이 있다.

본 연구에서는 설계변수 값의 초기값과 최대값을 결정하기 위한 지식기반으로 설계변수의 활성 규칙군과 제한값 규칙군으로 대별한다.

설계변수 값의 활성 규칙군은 다음과 같다.

$$\begin{aligned} \text{if } y(k) > (1 + e^*) \cdot r(k), \text{ then } \nu &= \nu - e(k) / ed \\ \text{if } y(k) < (1 - e^*) \cdot r(k), \text{ then } \nu &= \nu + e(k) / ed \end{aligned}$$

설계변수 ν 값의 활성 규칙군은 시스템의 출력과 추정된 매개변수와의 측정 오차가 클 경우 ($EPS > 0.01$, $EPS = y_r(k) - y(k)$)에만 적용되는 규칙군으로 시스템의 출력이 기준입력보다 과도한 경우와 부족한 경우로 나누어 설계변수 값을 적절히 조절한다. 여기서 $e(k)$ 는 제어 오차이고 ed 는 플랜트의 이득에 따라 정해지는 절분값으로 본 연구에서는 실험값으로 $20 \times (2.5 + \log(\hat{f}_3, \hat{f}_4, \hat{f}_5, \hat{f}_6))$ 의 절대값 중 큰 값; 여기에서 \hat{f}_i 는 RELS에서 구해지는 추정값)이면 모든 제어대상에 만족한다. 또한 e^* 는 Back-Up 제어기의 서어보제어 입력을 조절하는 규칙군에서 정해지는 값이다.

추정오차가 작을 경우($EPS < 0.01$)에는 설계변수의 제한값 규칙군이 적용된다. ν 는 0보다 작은 값이면 제어기 매개변수가 음수가 되므로 항상 양의 실수이다. 추정오차가 작을 경우에는 값을 그대로 유지한다. 그러나 구해진 설계변수 ν 의 값이 ν_{max} 값보다 큰 경우에는 다음 규칙군에 의해 최대값보다 작도록 유지되어야 한다.

$$\text{if } \nu < \nu_{min}, \text{ then } \nu = \nu_{min}$$

$$\text{if } \nu > \nu_{max}, \text{ then } \nu = \nu_{max}$$

따라서 ν 값은 항상 양수의 값으로 최대값 ν_{max} 를 갖는 설계변수이다. 여기서 ν_{min} 는 임의의 작은 양수값으로 여기서는 0.01이며 ν_{max} 는 수치형 알고리즘서 계산되는 값으로 다음과 같다.

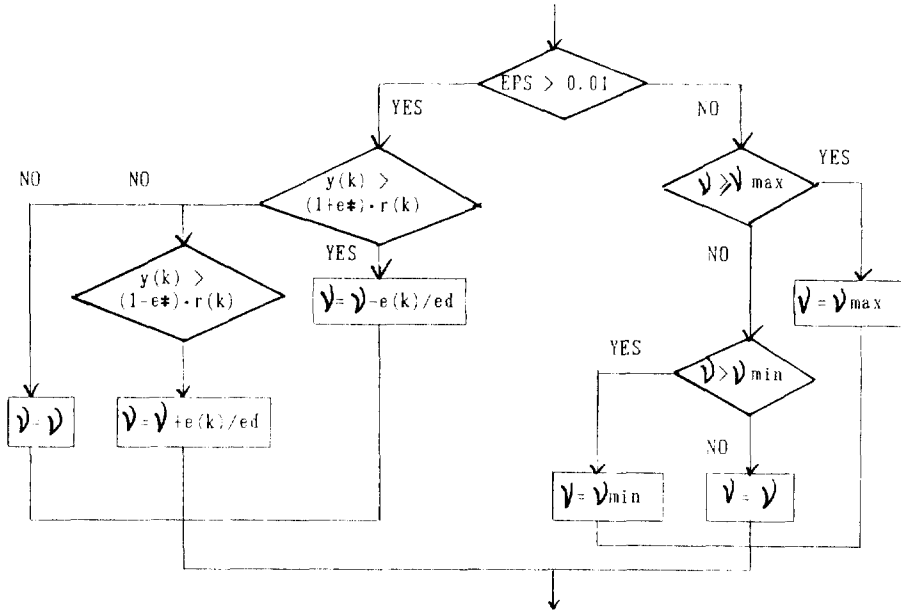


그림 4 Expert형 STC의 Scheduler
Fig. 4 The Scheduler of Expert STC

$v_{max} = (\hat{f}_3 + \hat{f}_4 + \hat{f}_5 + \hat{f}_6) / (\hat{f}_3, \hat{f}_4, \hat{f}_5, \hat{f}_6 \text{의 절대값 중 두번째 큰값})$

이상과 같은 지식 기반에 의해 설계변수 v 값을 조절하도록 Cameron과 Seborg의 STC 알고리즘을 수정 보완하면, 시스템의 매개변수의 변화나 지연시간의 변화에도 견실한 제어를 수행할 수 있으며, 초기값을 산정하지 않아도 되는 장점이 있다. 또한 비최소 위상시스템에도 적용가능하다. 따라서 본 연구에서 제안한 STC알고리즘은 다양한 시스템에 대해 유연한 제어 성능을 낼 수 있도록 고안된 Expert형 알고리즘이라 할 수 있다.

위에서 설명한 지식기반에 의한 Expert형 STC의 Scheduler는 그림4와 같은 구성을 이룬다.

3.3 불안정 상태지시기의 지식기반 및 Scheduler

불안정 상태 지시기는 시스템 출력을 감시하여 불안정의 상태가 감지되면 이를 경보하여 Expert형 Back-Up 제어기로 전환시켜 주기 위해 사용된다. 지시기 설계에서 중요한 점은 시스템의 안정 및 불안정을 구분하는 경계치의 설정이다. 이 값은 대상 시스템 및 부하에 따라 다르며, 논리적

알고리즘 보다는 Trial and Error 방법에 의해 결정되는 값이다.

본 연구에서는 다음과 같이 구성되는 불안정 상태지시기의 출력에 임계값 테스트를 적용하는 방법을 제시한다.

$$X_m(k) = |(y(k) - y(k-4))| / 2$$

불안정 상태지시기는 제어 대상의 출력을 감시

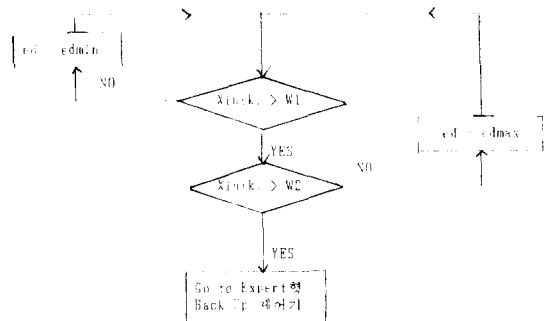


그림 5 불안정상태 지시기의 Scheduler
Fig. 5 The Scheduler of Instability Indicator.

하여, 지시기의 출력이 W1보다 큰 경우에는 Exper형 STC의 활성 규칙군에서 ed값을 최대값으로 보정한다. 그러나 제어 불능인 상황으로 지시기의 출력이 W2보다 큰 경우에는 Expert형 STC에서 Expert형 Back-Up 제어로 전환되어 Expert형 Back-Up 제어가 서어보제어와 조절 제어를 수행하여 $e(k) < 0.05 \cdot r(k)$ 이면 Expert형 STC로 재질환된다. 이때 Expert형 Back-Up 제어기는 동정 알고리즘을 병행 수행한다.

여기서 W1과 W2는 Back-up 제어기의 지식 기반에서 구해지는 양의 실수이나 본연구에서는 다음과 같이 선정하여 사용한다.

$$W1 = 0.4 \cdot r(k)$$

$$W2 = 2 \cdot W1$$

불안정 상태지시기의 규칙군은 다음과 같고 지시기반에 의한 scheduler는 그림5와 같다.

```

if  $X_{in}(k) < W1$ , then  $ed = ed_{min}$ 
if  $X_{in}(k) > W1$ , then  $ed = ed_{max}$ 
if  $X_{in}(k) > W2$ , then Goto Expert형 Back-up 제어기
    
```

본 연구에서 개발된 Expert형 제어기의 성능을 검토 확인하기 위하여 IBM-PC/AT를 사용하여 시뮬레이션을 수행하였다. 본 연구에서는 2장에서 이미 기술한 바와 같이 Turbo-Prolog를 이용하여 80개 내외의 규칙으로 지식기반을 형성하였고, 이를 Breadth-First기법에 의하여 Tracking하여 제어 입력을 선정하는 Scheduler를 개발하였다. 또한 Turbo-C에서는 기준 입력, 매개 변수, 시스템 출력 등의 수치들을 계산하도록 하여 이를

표 1 대상시스템 및 사용제어기에 따른 실험 분류
Table 1 The Experimental Classification by the system and controller.

구분	대 상	사용 제어기	결과
실험 I	시스템(1) + 시스템(2) + 시스템(1)	(1)	그림6
실험 II	시스템(1) + 시스템(3) + 시스템(5)	(1)	그림7
실험 III	시스템(1) + 시스템(3) + 시스템(5)	(1)	그림8
실험 IV	시스템(1) + 시스템(3) + 시스템(6)	(1)	그림9
실험 V	시스템(1) + 시스템(3) + 시스템(6)	(1) + (2)	그림10

(제어기(1) : Expert형 STC,
 제어기(2) : Expert형 Back-Up 제어기)

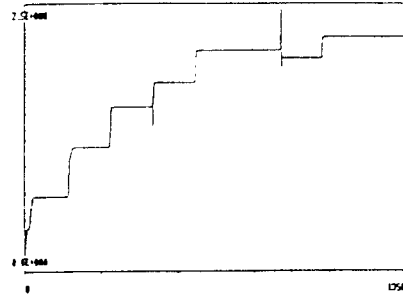


그림 6-a 실험 I의 설계 변수의 값
Fig. 6-a The value of design parameter in Experiment I

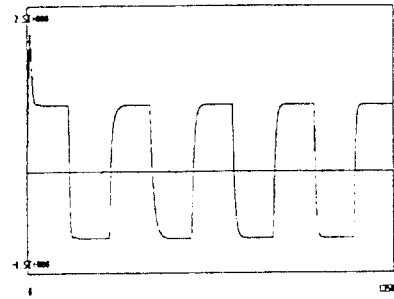


그림 6-b 실험 I의 제어입력
Fig. 6-b The control input in Experiment I

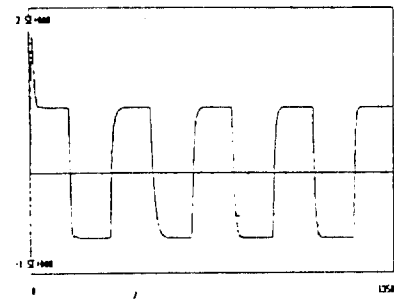


그림 6-c 실험 I의 출력
Fig. 6-c The output in Experiment I

Turbo-Prolog와 결합하였다. 실험에 사용된 조건은 다음과 같다.

- 조건1. 설계 매개 변수 값의 초기값은 0이다.
- 조건2. 시스템 매개 변수의 초기값은 모두 0이다.
- 조건3. RELS 동정 기법에서 초기 Covariance 값은 10이다.
- 조건4. 보형화 오차 및 잡음은 없다.

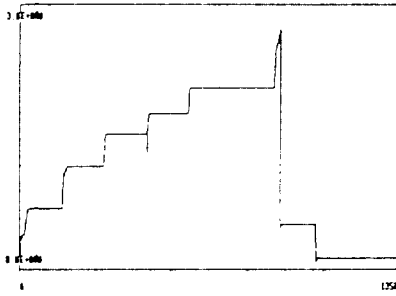


그림 7-a 실험 II의 설계변수의 값
 Fig. 7-a The value of design parameter in Experiment II

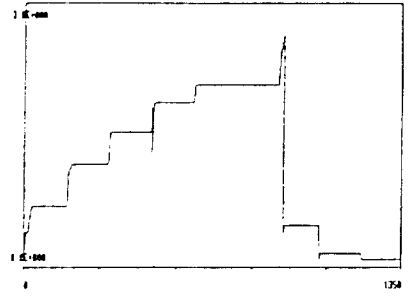


그림 8-a 실험 III의 설계변수의 값
 Fig. 8-a The value of design parameter in Experiment III

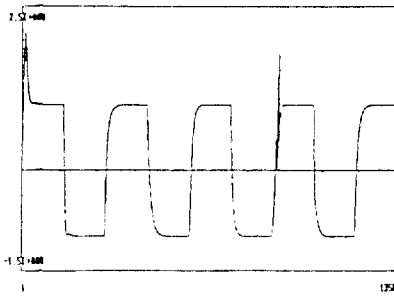


그림 7-b 실험 II의 제어입력
 Fig. 7-b The control input in Experiment II

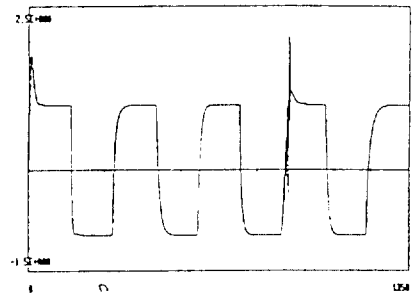


그림 8-b 실험 III의 제어입력
 Fig. 8-b The control input in Experiment III

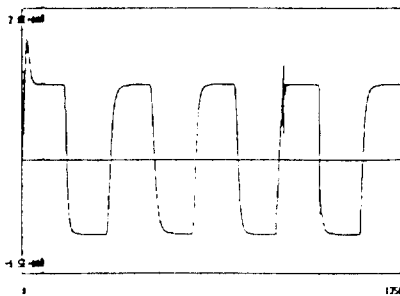


그림 7-c 실험 II의 출력
 Fig. 7-c The output in Experiment II

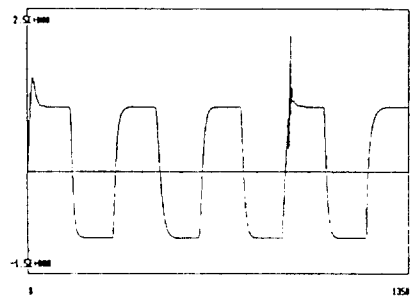


그림 8-c 실험 III의 출력
 Fig. 8-c The output in Experiment III.

실험에 사용된 제어 대상은 다음과 같은 차분 방정식으로 표시된 시스템을 사용하였고 표1과 같이 실험하였다. 사용된 기준입력은 150번의 샘플링 시간마다 기준입력이 바뀌는 구형파이다.

시스템 (1) : $y(k) = 1.5832 * y(k-1) - 0.5866 * y(k-2) + 0.028u(k-1) + 0.0234u(k-2)$

시스템 (2) : $y(k) = 1.5832 * y(k-1) - 0.5866 * y(k-2) + 0.028u(k-3) + 0.0234u(k-4)$

시스템 (3) : $y(k) = 1.5832 * y(k-1) - 0.5866 * y(k-2) + 0.0231u(k-1) + 0.0234u(k-2)$

시스템 (4) : $y(k) = 1.5832 * y(k-1) - 0.5866 * y(k-2) + 0.0231u(k-3) + 0.0745u(k-4)$

시스템 (5) : $y(k) = 0.857 * y(k-1) - 0.548 * y(k-2) + 0.381u(k-1) + 0.31u(k-2)$

시스템 (6) : $y(k) = 0.857 * y(k-1) - 0.548 * y(k-2) + 0.381u(k-3) + 0.31u(k-4)$

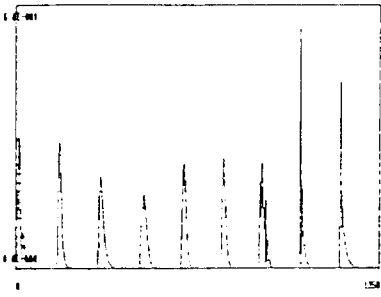


그림 9-a 실험 IV의 지시기의 출력
Fig. 9-a The output of indicator in Experiment IV

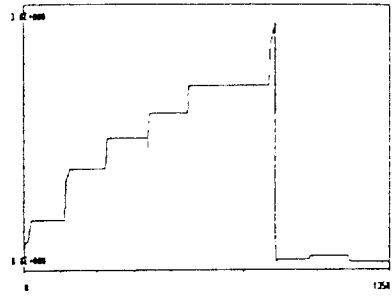


그림 9-b 실험 IV의 설계변수의 값
Fig. 9-b The value of design parameter in Experiment IV

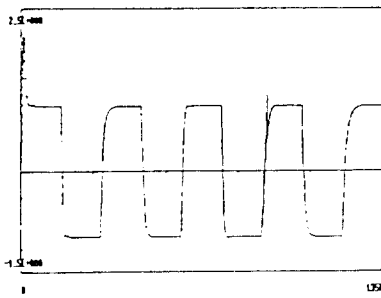


그림 9-c 실험 IV의 출력
Fig. 9-c The control input in Experiment IV

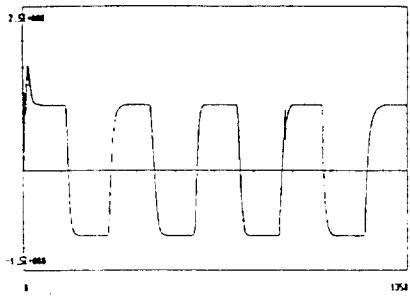


그림 9-d 실험 IV의 출력
Fig. 9-d The output in Experiment IV

실험 1은 시스템(1)이 $k=450$ 에서 지연 시간이 변하여 시스템(2)로 되고 $k=900$ 에서 다시 시스템(1)로 바뀌는 실험으로 실험 결과는 그림6과 같다. 실험II와 실험 III은 시스템(1)이 $k=450$ 에서 비최소 위상인 시스템(3)으로 되고 $k=900$ 에서 대상 시스템이 시스템(5)로 되는 실험으로 실험 II는 시스템 매개 변수의 초기값이 0인 경우로 실험 결과는 그림7과 같고, 실험 III은 모든 조건이 실험 II와 같으나 시스템 매개 변수의 초기값이 0.5인 경우로 그림8과 같다. 실험 I, 실험 II, 실험 III에 사용된 Expert STC는 매개 변수의 변화가 극심한 경우에는 불안정 상태 지시기에 의한 처방에도 불구하고 기준 입력이 바뀔 경우 시스템 출력의 심한 변화가 나타남을 알 수 있다. 그러나 오버슈트 현상은 심하지 않다.

실험 IV는 시스템(I)이 $k=450$ 에서 비최소 위상인 시스템(3)이 되고 $k=900$ 에서 시스템 매개 변수와 지연 시간이 동시에 변하는 시스템(6)이 되는 실험으로 Expert STC만을 사용하여 불안정 상

태 지시기의 조절에 의한 제어로 결과는 그림9과 같다. 이 경우에 Expert형 Back-up 제어기와 Expert형 STC를 동시에 사용한 실험 V는 초기시의 오버슈트와 매개변수의 변화가 극심한 경우 또는 매개 변수와 지연 시간이 동시에 변하는 조건에서도 그림10과 같이 만족할 만한 결과를 얻을 수 있다.

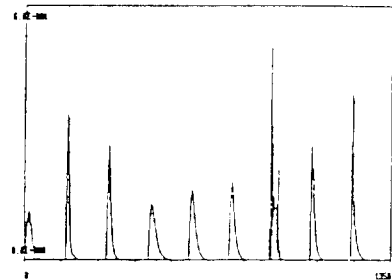


그림 10-a 실험 V의 지시기의 출력
Fig. 10-a The output of indicator in Experiment V

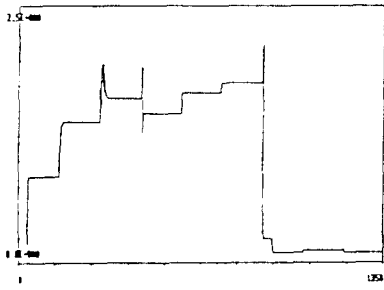


그림 10-b 실험 V의 설계변수의 값
 Fig. 10-b The value of design parameter in Experiment V

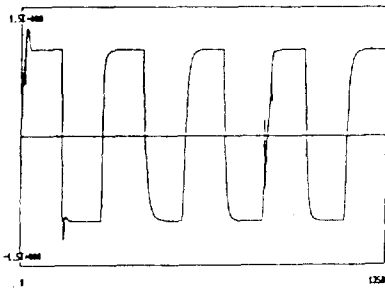


그림 10-c 실험 V의 제어입력
 Fig. 10-c The control input in Experiment V

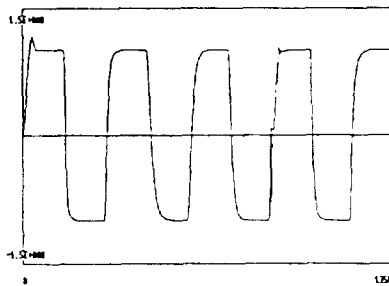


그림 10-d 실험 V의 출력
 Fig. 10-d The output in Experiment V

4. 결 론

본 연구에서는 Expert형 제어기법에 의한 자기동조 제어기를 구성하였다. 이 Expert형 제어기는 플랜트의 현재 상태를 감시하여 불안정한 상태를 지시하는 불안정 상태지시기, 이 불안정 상태지시기에 의해 동작 모드가 변경되는 Expert형 Back-up 제어기 및 Expert형 STC, 그리고 지식기

반 및 Inference Engine으로 이루어지는 전문가 시스템으로 구성된다. 제안된 Expert형 Back-up 제어기는 Sripada 등의 AI 제어기를 개선하여 (1) 서브제어와 조절 제어를 순차적으로 수행하며 (2) Fuzzy 논리에 의한 Membership 함수의 도입을 지양하여 기준입력과 제어오차 및 최대 제어 입력만으로 제어입력을 조정하여, 제어대상마다 Membership 함수를 정해야 하는 단점을 보완하였으며 (3) 각 규칙을 적용할 때 발생하는 불확실성을 방지하여 Breadth-First Tracking법으로 최대 7번, 최소 4번만에 제어 입력의 변화를 계산, 실행하여 제어 입력 결정시간을 단축하였다. 또한 Expert형 STC는 Cameron과 Seborg가 제안한 STC의 단점인 설계변수를 ν 를 자동 조정하도록 보완하여 (1) 설계변수 값의 초기값 및 최대값의 선정 (2) 지연시간의 변화에도 견실한 특성을 보였다. 그리고 불안정 상태지시기에 의한 동작모드의 선정으로 제어 대상의 특성 변동시 제어기 매개 변수의 재동조를 스스로 수행하므로써 단위 공정을 효율적으로 제어할 수 있음을 실험적으로 증명하였다. 그러나 전문가 시스템을 구성하기 위한 지식기반의 확충과 잡음이나 외란등이 존재하는 플랜트에 대한 필터의 연구가 계속되어야 할 것이다.

참 고 문 헌

- [1] F. Cameron and D.E. Seborg, "PID Structured Self-tuning Controller," INT, J, Control, vol. 38, no. 2, 1988.
- [2] R. Isermann, "Parameter Adaptive Control Algorithms—A Tutorial," Automatica, vol. 13, no. 5, 1988.
- [3] R. Isermann and K.H. Lachmann, "Parameter—Adaptive Control with Configuration Aids and Supervision Functions," Automatica, vol. 21, no. 6, 1985.
- [4] N.R. Sripada and D.G. Fisher, "Improved Least Squares Identification for Adaptive Controllers," ACC. vol. 3, 1987.
- [5] T.F. Chwee and H.R. Sirisena, "Self-tuning PID Controllers for Dead Time Processes," IEEE Trans. on Ind. Electronics, vol. 35, no. 1, 1988.
- [6] H.Kurz and W. Godecke, "Digital Parameter Adaptive Control of Process with Unknown

- Dead Time, "Automatica, vol. 17, no. 2, 1981.
- [7] C.A. Tsiligiannis and S.A. Svoronos, "Deterministic Convergence of a Clarke-Gawthrop Self-tuning Controller, "Automatica, vol. 22, no. 2, 1986.
- [8] P.S. Tuffs and D.W. Clarke, "Self-tuning Control of Offset : A Unified Approach," IEE Proceedings, vol. 132, no. 3, 1985.
- [9] J. Brunet, "On-Line Expert Supervision for Process Control," Proc. of 25th Conf. on Decision and Control, 1986.
- [10] J. Gertler and H.S. Chang, "An Instability Indicator for Expert Control," IEEE Control Systems Magazine, 1986.
- [11] K.E. Arzen, "Use of Expert Systems in Closed Loop Feedback Control," ACC., 1986.
- [12] K.J. Astrom, J.J. Anton, and K.E. Arzen, "Expert Control," Automatica, vol. 22, no. 5, 1986.
- [13] R.L. Moor, L.B. Hawkinson, and G. Knickerbocker, "Expert Control," ACC., 1985.
- [14] D. Ionescu, Ph. Lethebinh, and I. Trif, "Expert System for Computer Process Control Design," Proc. IEEE. Int. Sym. on Intelligent Control, 1987.
- [15] K.K. Gidwani, "The Role of Artificial Intelligence Systems in Process Control," Proc. ACC., 1985.
- [16] D.D. Freeman, "Artificial Intelligent Applications in Process Control," Proc. ACC., 1985.
- [17] W.F. Kaemmerer, and P.D. Christopherson, "Using Process Models with Expert Systems to Aid Process Control Operators," Proc. ACC., 1985.
- [18] M. Beaverstock, E.H. Bristol, and D. Fortin, "Expert Systems as a Stimulus to Improved Process Control," Proc. ACC., 1985.
- [19] T.L. Trankle, P. Sheu, and U.H. Rabin, "Expert System Architecture for Control System Design," Proc. ACC., 1986.
- [20] J. Jiang, and R. Doraiswmi, "Performance Monitoring in Expert Control Systems," IFAC 10th World Cong. on Automatic Control, 1987.
- [21] J. Jiang and R. Doraiswami, "Information Acquisition in Expert Control System Design Using Adaptive Filters," Proc. IEEE Int. Sym. on Intelligent Control, 1987
- [22] B. Porter, A.H. Jons, and C.B. Mckeowm, "Real-Time Expert Controllers for Plants with Actuator Non-linearities," IEEE Int. Sym. on Intelligent Control, 1987.
- [23] B.Porter, A.H. Jones, and C.B. Mckeowm, "Real-Time Expert Tuners for PI Controllers" IEEE, Proc. vol. 134, July. 1987.
- [24] N.R. Sripada, D.G. Fisher, and A.J. Morris, "AI Applications for Process Regulation and Servo Control," IEEE. Proc. vol. 134, July 1987.
- [25] N.R. Sripada, D.G. Fisher, and A.J. Morris, "Application of Expert Systems to Process Problems," Energy Processing, 1985.
- [26] S. Tjokro and S.L. Shah, "Adaptive PID Control," Proc. ACC., 1986.
- [27] R. Ortega and R. Kelly, "PID Self-Tuners Some Theoretical and Practical Aspects," IEEE Trans. Ind. Elec. vol. IE-31, no. 4, 1984,