

PID 제어기의 자동 계수 조정에 관한 연구

박종현 김경기
한양대학교 전자공학과

A study on the parameter self adjusting
method of the PID controller

*
Jong-Hyun Park, Kyung-Gi Kim
Dept. of Electronic Eng. of Hanyang Univ.

The characteristic of a control system is apt to be changed by environment and load condition.

This paper treats the algorithm for getting more optimal parameter by itself in PID control system with micro-processor.

1. 서론

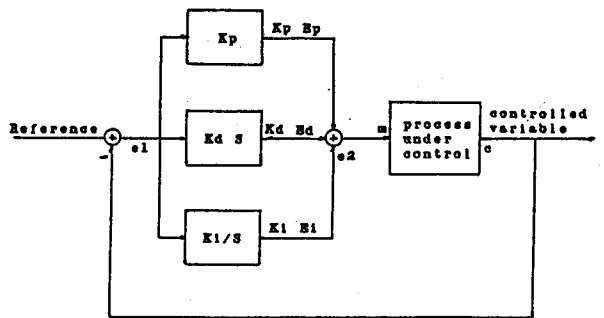
공정제어산업에서 컴퓨터 제어 기술은 날로 그 이용도가 증가하고 있다. 컴퓨터 제어는 하나의 프로세서가 여러개의 제어물 한꺼번에 처리하고 서로의 관련처리가 가능하기 때문에 많은 잇점이 있다. 이와 더불어 똑같은 기기로 제어과정을 프로그램 만들 변경함으로 자유로이 변경할 수 있기 때문에 그 최적점의 값을 찾기 쉽다. 수년동안 이러한 프로세스에 PID 제어기가 유용되어 왔다. [3, 4, 5]

이러한 PID 제어기는 선형제어와는 다른 성질을 나타내며 샘플링 시간에 따라 전체 제어계의 제어특성이 변화하게 된다. 또 제어대상계의 정확한 특성 파악은 사실상 어려운 것이며 이 특성 또한 주위환경, 부하 변동등의 원인에 의하여 많은 변화를 나타낸다. 그러므로 실제 응용에서는 최적 계수값을 실험을 통하여 얻게되는 경우가 많다. [2]

본 논문에서는 특성을 정확히 알 수 없는 임의의 시스템에 PID 제어기를 적용하여 평가함수가 최소가 되는 계수로 접근해가는 알고리즘을 제시하고 이의 시뮬레이션을 하였다.

2. PID 제어기

PID 제어기는 제어계에 비례(Proportional), 적분(Integral), 미분(Derivative) 제어를 가하는 것으로 <그림1>과 같은 구성으로 되어있다. [5]



<그림1> PID 제어계

<Fig1> PID control loop

<그림1>의 PID 제어기를 마이크로 프로세서로 구성하였을 경우 출력 e2를 Discrete Time 형태로 표현하면,

$$e2(k) = Kp Ep(k) + Ki Ei(k) + Kd Ed(k) \quad (1)$$

단)

$$Ep(k) = e1(k)$$

$$Ei(k) = Ei(k-1) + T e1(k)$$

$$Ed(k) = [e1(k) - e1(k-1)] / T$$

이를 Direct Programming법을 이용하여 State Model로 표현하면[5]

$$\begin{bmatrix} X1(k+1) \\ Xd(k+1) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} X1(k) \\ Xd(k) \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \end{bmatrix} e1(k) \quad \dots (2)$$

$$e2(k) = [K1 \ T \ -Kd/T] \begin{bmatrix} X1(k) \\ Xd(k) \end{bmatrix} + [Kp + K1 \ T + Kd/T] e1(k) \quad \dots (3)$$

3. PID 제어기의 최적화 알고리즘

최소편차제어를 위한 평가함수를

$$J = \int_0^{Tf} |e| dt \quad \dots (4)$$

이라 정의하면 이의 Discrete form은

$$J(kT) = \sum_{n=0}^{k-1} |e(n)T| \quad \dots (5)$$

이라고 할 수 있다.

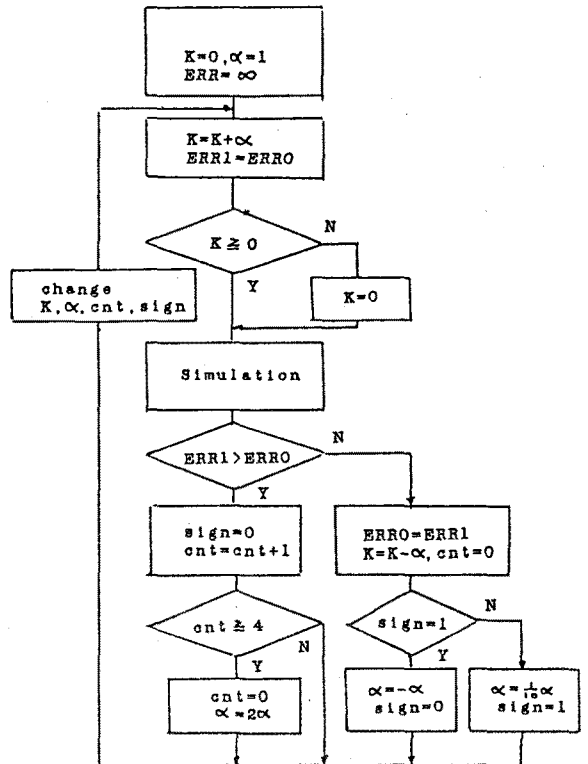
이 평가함수 J(kT)의 값이 최소가 되도록 step 입력을 제어 계에 가하고 Kp, K1, Kd의 값을 결정해 나간다. 이때의 Kp, K1, Kd의 값은 일정한 논리법칙에 의해 결정하고 이의 Simulation 후 평가지수의 값이 이전보다 작아지면 그 값을 선정 하고 보다 커지면 다른값을 찾는다. 초기 계수는 Kp=0, K1=0, Kd=0 에서 시작 하며 이의 누적 error범위 내에서 각 계수를 선정 하기 때문에 자기발진이 없다면 선정된 계수에 대한 System은 항상 안정하다.

<그림2>에 계수 자동 조정 Algorithm에 대한 flowchart 를 표시하였다. 여기서는 선택된 계수의 증분 이고 sign, cnt는 각 Flow에 대한 Flag 및 Count 이다.

4. SIMULATION

계수 자동 조정 알고리즘의 Simulation 을 위하여 다음의 전달함수를 갖는 System을 Test model로 채용 하였다.

$$G(s) = \frac{1}{20s + 12s + 1} \quad \dots (6)$$



<그림2> 계수 자동 조정 FLOWCHART
<Fig2> Flowchart of the Parameter Self Adjusting

이의 Matrix form은

$$\begin{bmatrix} x1 \\ x2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -1/20 & -3/5 \end{bmatrix} \begin{bmatrix} x1 \\ x2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1/20 \end{bmatrix} m(t) \quad \dots (7)$$

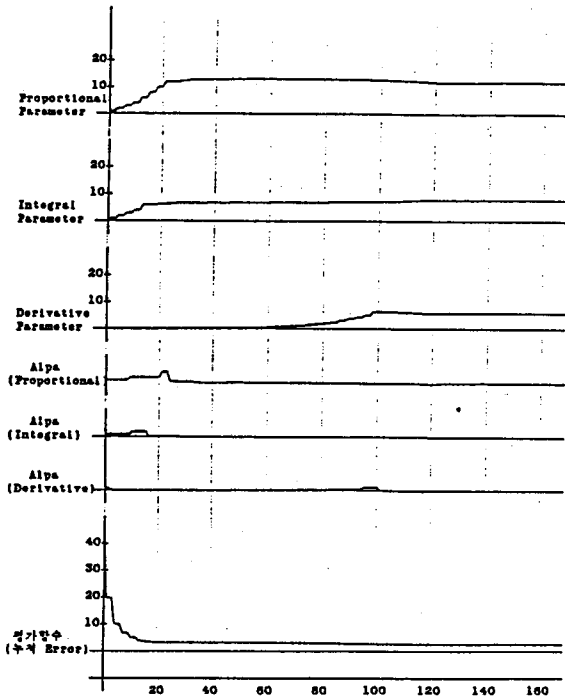
$$c(t) = x1 \quad \dots (8)$$

이를 Computer program을 위하여 Time-domain 의 Transition matrix를 구하면 다음과 같이된다.

$$\begin{bmatrix} x1(k+1) \\ x2(k+1) \end{bmatrix} = \begin{bmatrix} \frac{5}{4} e^{-T/2} - \frac{1}{4} e^{-T/10} & \frac{5}{2} e^{-T/2} - \frac{5}{2} e^{-T/10} \\ -\frac{1}{8} e^{-T/2} + \frac{1}{8} e^{-T/10} & -\frac{1}{4} e^{-T/2} - \frac{5}{4} e^{-T/10} \end{bmatrix} \begin{bmatrix} x1(k) \\ x2(k) \end{bmatrix} + \begin{bmatrix} \frac{1}{20} - \frac{1}{10} e^{-T/2} + \frac{1}{80} e^{-T/10} \\ \frac{1}{160} e^{-T/2} - \frac{1}{160} e^{-T/10} \end{bmatrix} m(k) \quad \dots (9)$$

$$c(k) = x1(k) \quad \dots (10)$$

Simulation 방법은 먼저 Reference 값과 제어된 값 $c(k)$ 와의 차 (error)를 계산하고 이를 (2), (3) 식에 대입하여 $e_2(k)$ 를 구한다. 이값을 (9)식의 $m(k)$ 값으로 하고 (9)식에 의해 $c(k)$ 을 구한다. 이 과정을 Simulation 기간동안 Sampling time 에 맞춰 반복한다. 본 논문에서의 Sampling time은 1[sec]로 하였다.

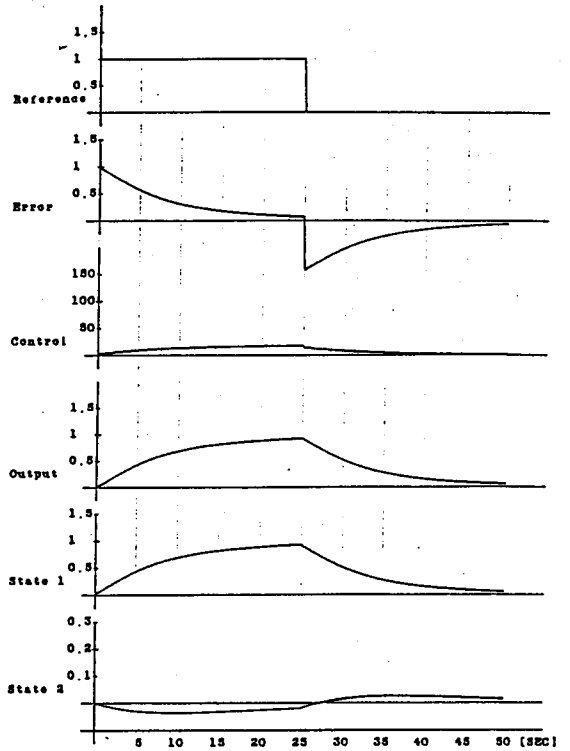


<그림3> SIMULATION에 따른 각 계수의 변화
<Fig3> The variations of each parameter by simulation

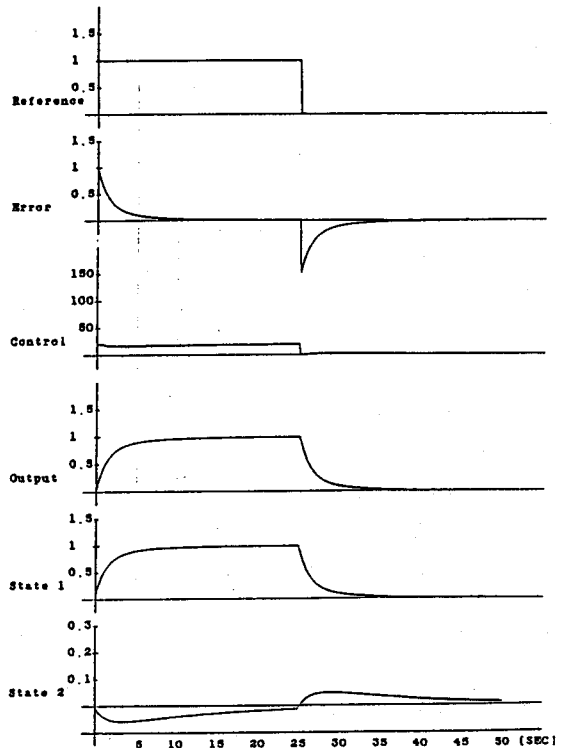
<그림3>에 Simulation의 반복횟수에 따른 각 계수(K_p, K_i, K_d)의 변화를 나타냈고 각 계수의 증분 Alpha를 표시하였다. <그림3>에서 보인 것과 같이 평가 함수 $J(kT)$ 의 값은 반복 횟수에 따라 점차 감소하여 일정한 값에 이른다. 이때 평가함수의 감소는 처음 급격히 감소 하다가 조금씩 감소되는데 대상 System에 따라 감소형태는 달라 진다.

<그림4>, <그림5>, <그림6>에 각 시뮬레이션 에서의 STEP 입력에 대한 출력 특성을 나타냈다. 이때 적용된 각 계수의 값과 평가함수는 다음과 같다.

<그림4> 8 번째 SIMULATION 후
 $K_p = 2, \quad K_i = 2, \quad K_d = 0$
 $J = 19.983$



<그림4> STEP 입력에 대한 응답
<Fig4> Response to step input



<그림5> STEP 입력에 대한 응답
<Fig5> Response to step input

<그림5> 50 번째 SIMULATION 후

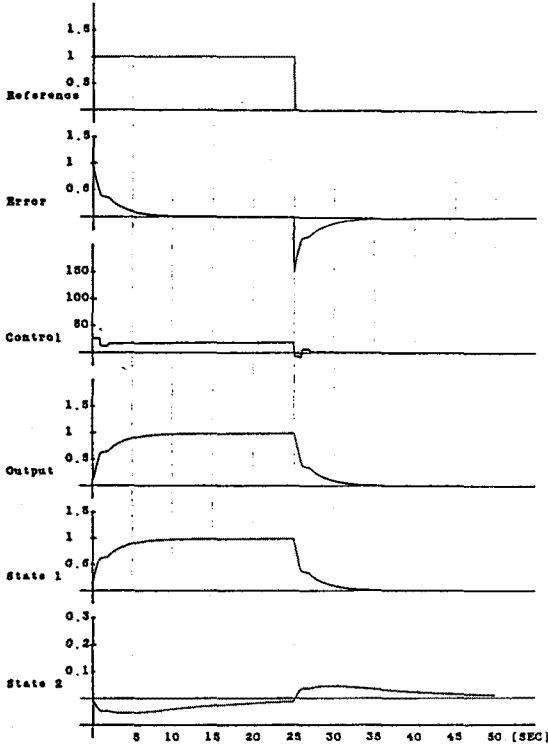
$K_p = 13.04, K_i = 6.8, K_d = 0.36$

$J = 6.586$

<그림6> 100 번째 SIMULATION 후

$K_p = 12.7, K_i = 7.26, K_d = 6.36$

$J = 8.592$



<그림6> STEP 입력에 대한 응답

<Fig6> Response to step input

5. 결론

종래의 PID Controller의 Parameter 조정법은 외란에 대한 응답시간 최소 및 목표치 변화에 대한 응답시간 최소에 초점을 두고 결정하는 방법인데 반해 본 논문에서는 평가함수를 도입하여 평가 함수에 따른 Optimize 한 Parameter로 조정하는 방식이다.

수학적 계산에 의한 결정법은 대상 System의 정밀한 분석이 요구되어 지며 실험을 통하여 다시 조정하지 않으면 안된다. 본 논문에서 제안한 방법은 대상 System의 정밀한 분석없이 적용할 수 있고 이러한 Algorithm을 Controller에 내장함으로 수시로 Optimize한 값을 구할 수 있다.

앞으로 실제 실험을 통하여 적용정도 분석 및 보다 나은 Algorithm의 연구가 요구된다.

참고 문헌

1. PETER J. GAWTHROP "Self-tuning PID Controllers : Algorithms and Implementation" IEEE Trans. on Automatic control vol AC-31 March 1986
2. CHARLES L. PHILLIPS, JOHN M. PARR "Robust Design of a Digital PID Predictor Controller" IEEE Trans. on Industrial Electronics vol IE-31 No 4 November 1984
3. TAKESHI TSUCHIYA "Improved Direct Digital Control Algorithm for Microprocessor Implementation" IEEE Trans. on Automatic Control vol AC-27 No 2, April 1982.
4. Stanley M. Shinnars "Modern Control System Theory and Application"(Book) pp 76 - 79, 1979.
5. JAMES A. CADZOW HINRICH R. MARTENS "Discrete-time and Computer Control System" (Book) pp 100 - 107 1970.
6. Karl J. Astrom Bjorn Wittenmark "Computer Controlled System : Theory and Design" (Book) pp 100 - 193.
7. PLEXUS "Sys3 UNIX Programmer's Manual" (Book) vol 1, vol 2.
8. A-TECH "ATLAS-9X Programmer's Manual" (Book) 1987.