

## GPC기법을 이용한 자기동조 PID제어기 설계

윤 강 섭 부산대학교 대학원 정밀기계공학과  
이 만 형 부산대학교 제어기계공학과  
부산대학교 기계기술연구소 및 NSDM-ERC 연구원

### Abstract

PID control has been widely used for real control systems. Further, there are much reseaches on control schemes of tuning PID gains. However, there is no results for discrete-time systems with unknown time-delay and unknown system parameters. On the other hand, Generalized predictive control has been reported as a useful self-tuning control technique for systems with unknown time-delay. So, in this study, based on minimization of a GPC criterion, we present a self-tuning PID control algorithm for unknown parameters and unknown time-dclay system. A numerical simulation was presented to illuatrte the effectiveness of this method.

### 1. 서 론

PID제어기는 다른 제어기에 비해 제어기의 구조가 간단하고 제어파라메타의 물리적인 의미를 이해하기 쉽다는 장점을 가지고 있으며, 현재도 산업현장에서는 많이 이용되고 있다. 또한, PID 이득선정에 관한 연구도 많이 진행되어 왔다[1-5]. 특히 미지의 파라메타를 가진 시스템에 대해서 자기동조(self-tuning)기법을 이용한 PID 제어기에 관한 많은 연구가 있었다. 이러한 자기동조 PID 제어방법은 시스템의 정보가 불충분하거나 시스템의 파라메타가 변화하는 경우에 대해서 적용할 수 있도록, 시스템의 파라메타를 추정하여 제어기의 이득을 변화시키는 방법이다. 그러나 자기동조 PID 제어기는 시스템의 정보, 특히 시스템의 차수, 시간지연에 대한 정보가 부정확할 경우는 문제점이 발생한다. 그래서 미지의 시스템 차수나 시간지연을 가진 시스템에 대해서도 유용한 GPC(Generalized Predictive Control) 기법[6-10]을 이용하여 PID의 이득을 선정할 수 있는 제어기를 구성해 보고자 한다.

미지의 시간지연을 가진 시스템에 대해서 GPC기법을 PID 이득선정에 이용하기 위해서, 우선 속도형 PID 제어칙과 GPC 가 격함수의 최소화에 근거한 제어칙을 구하고, PID 제어구조가 GPC구조의 특별한 경우에 해당된다는 성질을 이용하여 PID 이득조정방법을 제안하여, 자기동조제어시스템의 설계에 적용하였다. 그리고 이 방법의 효용성을 보이기 위해서 간단한 시뮬레이션 예제를 보였다.

### 2. 자기동조 PID 제어기

제어대상 플랜트로 ARIMA(Auto-Regressive and Integrated Moving-Average)모델의 단일입출력 시스템을 고려하자.

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

여기서,  $u(k)$ 는 입력,  $y(k)$ 는 출력,  $A(z^{-1}) = 1 + a_1z^{-1} + \dots + a_nz^{-n}$ ,  $B(z^{-1}) = b_0 + b_1z^{-1} + \dots + b_nz^{-n}$ ,  $z^{-1}$ 는 shift operator,  $\Delta$ 는 differential operator( $\Delta = 1 - z^{-1}$ ),  $\xi(k)$ 는 외란으로 백색 가우시안 프로세스이다. 만일 플랜트가 시간지연을 가진다면  $B(z^{-1})$ 의 첫번째 요소는 0이다. 시스템의 시간지연  $d$ 은 미지이고  $0 \leq d \leq nb$ 에 존재한다고 가정한다. 그리고, 기준입력  $\{r(k)\}$ 는 계단입력으로 주어진다고 가정하자. 플랜트 모델을 ARIMA 모델로 선택한 이유는 내부적으로 적분동작을 가진 제어기를 유도하기 위해서이다.

#### 2.1 속도형 PID 제어칙

연속시간에서 표준형 PID의 식은 다음과 같다.

$$u(t) = K \left[ e(t) + \frac{1}{T_i} \int_0^t e(\tau) d\tau + T_d \frac{de(t)}{dt} \right] \quad (2)$$

여기서,  $e(t)$ 는 오차신호로서  $e(t) = r(t) - y(t)$ 이고,  $r(t)$ 는 기준입력,  $y(t)$ 는 시스템의 출력이다. 이산시간에서의 속도형 PID로 나타내기 위해서, 식(2)를  $t = kT_s$ ,  $t = (k-1)T_s$ 에서 이산시간으로 등가하면,

$$u(k) = K \left\{ e(k) + \frac{1}{T_i} \int_0^{kT_s} e(\tau) d\tau + \frac{T_d}{T_s} [e(k) - e(k-1)] \right\} \quad (3)$$

$$u(k-1) = K \left\{ e(k-1) + \frac{1}{T_i} \int_0^{(k-1)T_s} e(\tau) d\tau + \frac{T_d}{T_s} [e(k-1) - e(k-2)] \right\} \quad (4)$$

여기서,  $T_s$ 는 샘플링주기이다. 식(3)에서 식(4)을 빼면,

$$u(k) - u(k-1) = K \left\{ e(k) - e(k-1) + \frac{1}{T_i} \int_{(k-1)T_s}^{kT_s} e(\tau) d\tau + \frac{T_d}{T_s} [e(k) - 2e(k-1) + e(k-2)] \right\} \quad (5)$$

$$\Delta u(k) = K \left\{ \Delta e(k) + \frac{T_s}{T_i} e(k) + \frac{T_d}{T_s} \Delta^2 e(k) \right\} \quad (6)$$

만일, 기준입력이 계단입력이라고 가정할 경우,  $\Delta e(k) = -\Delta y(k)$ ,  $\Delta^2 e(k) = -\Delta^2 y(k)$ 로 둘 수 있으므로, 속도형 PID (I-PD구조)를 나타낼 수 있다.

$$\Delta u(k) = K [e(k) - \{\Delta K_P + \Delta^2 K_D\} y(k)] \quad (7)$$

여기서,  $K_P=K$ ,  $K_I=K \frac{T_s}{T_i}$ ,  $K_D=K \frac{T_d}{T_s}$  이다.

미지의 파라미터와 미지의 시간지연을 가진 시스템에 대해서, GPC기법을 이용하여 PID이득을 선정하기 위해서, 다음으로 속도형 PID와 관계를 가질수있는 GPC 법칙을 유도하자.

## 2.2 GPC 법칙의 유도

GPC 기법에 사용되는 가격함수는 다음의 형태이다.

$$J=E\left\{\sum_{k_1}^{N_1}[y(k+j)-r(k+j)]^2+\sum_{k_2}^{N_2}Q_j(z^{-1})(\Delta u(k+j-1))^2\right\} \quad (8)$$

여기서,  $N_1$ 은 minimum costing horizon,  $N_2$ 는 maximum costing horizon이다. 시스템의 지연시간을 모른다는 가정하에  $N_1=1$ ,  $N_2=M$ 으로 두고, 입력이 계단 입력으로 일성하다는 가정하에  $r(k+j)=r(k)$ 로 두면, 가격함수는 다음과 같다.

$$J=E\left\{\sum_{k_1}^M[y(k+j)-r(k)]^2+\sum_{k_2}^M Q_j(z^{-1})(\Delta u(k+j-1))^2\right\} \quad (9)$$

여기서, M은 예측범위(prediction range)이고, 제어입력 가중요소  $Q_j(z^{-1})$ 는 다음과 같이 둔다.

$$Q_j(z^{-1})=q_{j,0}+q_{j,1}z^{-1}+\dots+q_{j,n}z^{-n} \quad (10)$$

GPC법칙은 가격함수가 최소화되도록하는 입력신호  $u(k)$ 를 결정하는 문제이다.  $u(k)$ 를 구하는 절차는 다음과 같다.

우선, j-step ahead predictor  $y(k+j)$ 를 유도하기 위해서, 다음의 Diophantine 방정식을 고려하자.

$$1=E_j(z^{-1})\Delta A(z^{-1})+z^{-j}F_j(z^{-1}) \quad (11)$$

여기서,  $E_j(z^{-1})=1+e_{j,1}z^{-1}+\dots+e_{j,n}z^{-n}$ ,

$$F_j(z^{-1})=f_{j,0}+f_{j,1}z^{-1}+\dots+f_{j,m}z^{-m}$$

이고,  $E_j$ 와  $F_j$ 는  $A(z^{-1})$ 와 예측구간(prediction interval) j에 의해서 유일하게 결정되는 다항식이다.  $E_j$ ,  $F_j$ 는 순환적으로 구할 수 있다[6]. 식(11)의 양변에  $E_j(z^{-1})\Delta z^j$ 를 곱하면,

$$E_j\Delta y(k+j)=E_jB\Delta u(k+j-1)+E_j\zeta(k+j) \quad (12)$$

식(11)으로부터  $E_j\Delta=1-z^{-j}F_j$ 를 대입하면,

$$y(k+j)=E_jB\Delta u(k+j-1)+F_jy(k)+E_j\zeta(k+j) \quad (13)$$

여기서,  $E_j(z^{-1})$ 는 j-1차이므로, 잡음성분은 모두 미래의 성분이고, 시간 k까지의 측정된 출력과  $u(k+i)$ ,  $i>1$  가 주어질때 optimal predictor는 다음과 같이 나타낼 수 있다.

$$\hat{y}(k+j|k)=G_j(z^{-1})\Delta u(k+j-1)+F_j(z^{-1})y(k) \quad (14)$$

여기서,  $G_j=E_jB=g_{j,0}+g_{j,1}z^{-1}+\dots+g_{j,m}z^{-m}$  ( $j=1,2,\dots,M$ )이다. 식(14)에서, 시간 k에서 알고 있는 성분을  $f(t+j)$ 라 두면,

$$\begin{aligned} f(k+j) &= z^{-1}[G_j(z^{-1})-z^{-(j-1)}g_{j,1}-\dots-z^{-1}g_{j,1}-g_{j,0}]\Delta u(k) \\ &\quad +F_j(z^{-1})y(k) \\ &= \sum_{i=1}^m g_{j,i}z^{-(i-1)}\Delta u(k-1)+F_j(z^{-1})y(k) \end{aligned} \quad (15)$$

식(14)를 시간 k에서 미지의 성분과 알고있는 성분  $f(t+j)$ 으로 분리하여 표현하면 다음과 같다.

$$\hat{y}(k+j|k)=[g_{j,0}+z^{-1}g_{j,1}+\dots+z^{-(j-1)}g_{j,j-1}]\Delta u(k+j-1)+f(k+j) \quad (16)$$

식(16)은 전체 예측구간에 대해서 벡터식으로 표현하면,

$$\hat{\underline{y}}=G\hat{\underline{u}}+f \quad (17)$$

여기서,  $\hat{\underline{y}}=[\hat{y}(t+1|k), \hat{y}(t+2|k), \dots, \hat{y}(t+M|k)]^T$

$$\hat{\underline{u}}=[\Delta u(t), \Delta u(t+1), \dots, \Delta u(t+M-1)]^T$$

$$\underline{f}=[f(t+1), f(t+2), \dots, f(t+M)]^T$$

이고, 행렬 G는  $G_j(z^{-1})$ 의 계수로 구성된다.

$$G=\begin{bmatrix} g_{1,0} & 0 & 0 & \dots & 0 \\ g_{2,1} & g_{2,0} & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ g_{M,M-1} & g_{M,M-2} & \dots & g_{M,0} & 0 \end{bmatrix} \quad (18)$$

여기서,  $G_j=E_jB=B[1-z^{-j}F_j]/\Delta A$ 이므로,  $G_j(q^{-1})$ 는 폴랜드에 대한 스템응답의 Z변환에서 처음 j개의 항이 되므로, 다항식  $G_j(q^{-1})$ 에 관계없이  $g_{ij}=g_i$  ( $j=1,2,\dots,i$ )이다.

식(17)의 벡터식을 이용하여 가격함수를 다시 나타내면,

$$J=E\left\{(\underline{G}\hat{\underline{u}}+\underline{f}-\underline{r})^T(\underline{G}\hat{\underline{u}}+\underline{f}-\underline{r})+\hat{\underline{u}}^T Q(z^{-1})\hat{\underline{u}}\right\} \quad (19)$$

여기서,  $\underline{r}=[r(k), r(k), \dots, r(k)]^T$

$$Q(z^{-1})=\text{diag}\{Q_j(z^{-1})\}=Q_0+Q_1z^{-1}+\dots+Q_Mz^{-M}$$

한편, 행렬  $Q(z^{-1})$ 를 이루고 있는 제어입력 가중요소  $Q_j(z^{-1})$ 는 다음과 같이 가정한다.

$$Q_1(z^{-1})=q_{1,0}+q_{1,1}z^{-1}+\dots+q_{1,n}z^{-n}$$

$$Q_j(z^{-1})=q_{j,0} \quad (j=2,3,\dots,M) \quad (20)$$

가격함수 식(19)는  $\hat{\underline{u}}$ 의 2차함수이므로, 미래의 제어입력에 대한 구속조건이 없다는 가정하에서 가격함수를 최소화하는 제어중분 벡터  $\hat{\underline{u}}$ 는 다음과 같다.

$$(G^T G+Q)\hat{\underline{u}}=G^T(\underline{r}-\underline{f}) \quad (21)$$

$$(G^T G+Q_0)\hat{\underline{u}}+Q_r\hat{\underline{u}}=G^T(\underline{r}-\underline{f}) \quad (22)$$

여기서,  $Q_r=Q_1z^{-1}+\dots+Q_Mz^{-M}$ ,  $Q_r$ 의 차원은  $M \times M$ 이고, 식(20)의 가정에 의해서,  $q_{r,1,1}=q_{1,1}z^{-1}+q_{1,2}z^{-2}+\dots+q_{1,n}z^{-n}$ 이고, 나머지 요소는 모두 0이다. 식(22)의 좌변 두번째 요소를 R이라 두면,

$$\begin{aligned} R=Q_r\hat{\underline{u}} &= [(q_{1,1}z^{-1}+q_{1,2}z^{-2}+\dots+q_{1,n}z^{-n})\Delta u(k), 0, 0, \dots, 0]^T \\ &= [R_1, 0, 0, \dots, 0]^T \end{aligned} \quad (23)$$

여기서,  $R_1=\sum_{i=1}^n q_{1,i}\Delta u(k-i)$ 이다. 식(23)을 식(22)에 대입하여 정리하면,

$$\hat{\underline{u}}=(G^T G+Q_0)^{-1}G^T[(\underline{r}-\underline{f})-(G^T)^{-1}R] \quad (24)$$

제어중분벡터  $\hat{\underline{u}}$ 의 첫번째 요소는  $\Delta u(k)$ 이므로, 현재의 제어입력  $u(k)$ 는 다음과 같다.

$$u(k)=u(k-1)+\underline{p}^T(\underline{r}-\underline{f})+\underline{p}^T(G^T)^{-1}R \quad (25)$$

여기서,  $\underline{p}^T$ 는 행렬  $(G^T G+Q_0)^{-1}G^T$ 의 첫번째 행벡터이므로,

$$\underline{p}^T=[p_1, p_2, \dots, p_M] \text{로 나타낸다.} \quad \text{식(25)를 정리하면,}$$

$$u(k)=u(k-1)+\sum_{j=1}^M p_j r(k)-\sum_{j=1}^M p_j f(k+j)+\frac{p_1}{g_{1,0}} R_1 \quad (26)$$

식(26)에 식(15)를 대입하면,

$$\begin{aligned} u(k) &= u(k-1)+\sum_{j=1}^M p_j r(k)-\sum_{j=1}^M p_j \left[ \sum_{i=1}^m g_{j,i}z^{-(i-1)}\Delta u(k-1) \right. \\ &\quad \left. +F_j(z^{-1})y(k) \right] + \frac{p_1}{g_{1,0}} R_1 \end{aligned} \quad (27)$$

$$u(k) = u(k-1) + \sum_{j=1}^M p_j r(k) - \sum_{j=1}^M p_j \sum_{i=1}^{nb} g_{j,i} z^{-i} \Delta u(k-1) + \sum_{j=1}^M p_j F_j(z^{-1}) y(k) + \frac{p_1}{g_{1,0}} \sum_{i=1}^l q_i \Delta u(k-i) \quad (28)$$

### 2.3 PID 제어칙과 GPC법칙의 결합

속도형 PID제어기의 식(7)을 다음과 같이 표현할 수 있다.

$$u(k) = u(k-1) + K_I r(k) - [(K_P + K_I + K_D) - (K_P + 2K_D)z^{-1} + K_D z^{-2}] y(k) \quad (29)$$

한편,  $l = nb$ ,  $q_{1,i} = \frac{g_{1,0}}{p_1} \sum_{j=1}^M p_j g_{j,i+1}$ 라 두면, GPC 법칙 식(28)에

서 우변 세번째항과 다섯번째항이 소거되어 다음과 같다.

$$u(k) = u(k-1) + \sum_{j=1}^M p_j r(k) + \sum_{j=1}^M p_j F_j(z^{-1}) y(k) \quad (30)$$

플랜트의  $A(z^{-1})$ 의 차수  $na$ 를 2차로 가정하면,  $F_j(z^{-1})$ 는 항상 2차이므로, 식(29)과 식(30)을 비교하여 다음과 같은 식을 얻을 수 있다.

$$K_I = \sum_{j=1}^M p_j, \quad K_P + K_I + K_D = - \sum_{j=1}^M p_j f_{j,0}, \\ K_P + 2K_D = \sum_{j=1}^M p_j f_{j,1}, \quad K_D = - \sum_{j=1}^M p_j f_{j,2} \quad (31)$$

식(31)을 정리하면,

$$K_P = - \sum_{j=1}^M p_j [f_{j,1} + 2f_{j,2}], \quad K_I = \sum_{j=1}^M p_j, \quad K_D = \sum_{j=1}^M p_j f_{j,2} \quad (32)$$

식(32)에 의해서 PID 이득을 결정할 수 있다.

식(7)으로 부터 다음과 같이 나타낼 수 있다.

$$u(k) = u(k-1) + K_P [y(k-1) - y(k)] + K_I [r(k) - y(k)] + K_D [-y(k) + 2y(k-1) - y(k-2)] \quad (33)$$

이상에서 구한 GPC기법을 이용한 PID이득 선정 방법을 미지의 시스템 파라미터인 경우로 확장하기 위해서는 파라미터의 추정 필요하므로, 자기동조 제어기의 기본 구조를 따른다. 파라미터 추정기로서는 exponential forgetting을 가진 Recursive least-squares estimation 알고리즘[11]을 고려할 수 있다.

$$K(k) = P(k-1)\Phi(k)[\lambda I + \Phi^T(k)P(k-1)\Phi(k)]^{-1} \\ P(k) = [P(k-1) - K(k)\Phi^T(k)P(k-1)]\lambda \quad (34)$$

$$\hat{\theta}(k) = \hat{\theta}(k-1) + K(k)[y(k) - \Phi^T(k)\hat{\theta}(k-1)]$$

여기서,  $\hat{\theta}(k) = [a_1, \dots, a_{na}, b_0, b_1, \dots, b_{nb}]^T$ .

$$\Phi(k) = [-y(k-1), \dots, -y(k-na), u(k-1), \dots, u(k-nb-1)]^T$$

이상과 같이 GPC기법을 이용한 자기동조 PID 제어기 알고리즘을 구성할 수 있다. 그 계산 절차를 요약하면 다음과 같다.

- (1) 설계파라미터로서 예측구간 M, 제어입력가중행렬  $Q_0$ 를 결정
- (2) 추정기를 사용하여, 시스템파라미터  $\hat{a}_i(k)$ ,  $\hat{b}_i(k)$ 를 추정
- (3) 주어진  $A(z^{-1})$ 로부터,  $F_j(z^{-1})$ 와  $G_j(z^{-1})$ 를 계산
- (4)  $p$ 를 계산하여, PID 이득을 계산
- (5) 제어입력  $u(k)$ 를 계산
- (6) (2)-(5)를 반복

### 3. 시뮬레이션 및 고찰

제안된 방법의 효율성을 조사하기 위해서, 다음의 비최소위상 시스템에 대해서 시뮬레이션을 수행하였다.

$$G(s) = \frac{(s-50)(s-20)(s+25)}{10(s+20)(s+20)(s+10)} e^{-0.2s}, \quad T_s = 0.1(\text{sec})$$

이산시간에서 표현하면

$$y(k) = 0.4762y(k-1) - 0.04762y(k-2) + 0.1286u(k-3) + 0.3143u(k-4) + 0.03333u(k-5) + \xi(k)$$

여기서,  $\xi(k)$ 는 평균이 0이고, 표준편차가 0.005인 백색 가우시안 프로세스이다. 시간지연은 모른다는 가정하에, 시스템 파라미터의 추정을 위해서 다음과 같은 모델을 사용하였다.

$$y(k) = -a_1(k)y(k-1) - a_2(k)y(k-2) + b_0(k)u(k-1) + b_1(k)u(k-2) + b_2(k)u(k-3) + b_3(k)u(k-4) + b_4(k)u(k-5) + \xi(k)$$

시뮬레이션 조건은 다음과 같다.

- (1) 기준입력 : 0-50(step) : 0  
51-100(step) : 5  
101-150(step) : 10  
151-200(step) : 0
- (2) 설계파라미터 : M : 10

$$Q_0 : 0.5 I_{M \times M}, 1.0 I_{M \times M}, 5.0 I_{M \times M}$$

위와 같은 조건으로 시뮬레이션을 수행하여 기준입력과 출력을 Fig. 1 에, 제어입력을 Fig. 2에 나타내었다. 그리고, Fig. 3은  $Q_0 = 1.0 I_{M \times M}$ 인 경우의 PID이득을 나타내었다. 그리고, 세가지 경우에 대한 PID이득의 수렴값을 Table.1 에 나타내었다.

Table. 1 PID gains for weighting factor

weighting factor	Kp	Ki	Kd	K	Ti	Td
0.5	0.28	0.46	0.021	0.28	0.12	0.015
1.0	0.21	0.29	0.039	0.21	0.15	0.037
5.0	0.07	0.10	0.012	0.07	0.0014	0.034

GPC알고리즘은 Receding horizon strategy를 사용하고 있다. 만일 추정과정중에 시간지연이 정확히 추정된다면(즉,  $b_0 \approx 0$ 인다면),  $N_1 = 1$ 로 두었기 때문에, 제어입력중분은 제어입력가중행렬  $Q_0$ 에 의해서만 결정되므로, 정확한 제어를 기대할 수 없다. 이 경우에 제어기 구조를 변경하여야 하지만, 본 연구에서는 식(25)의  $p^T$ 를 행렬  $(G^T G + Q_0)^{-1} G^T$ 의 첫번째 행벡터만 취하는 것이 아니라, 첫번째 행벡터 부터 예상되는 지연시간보다 1이 많은 행벡터까지의 합을  $p^T$ 로 두어 사용하여 양호한 결과를 얻을 수 있었다.

시뮬레이션 결과를 살펴보면, 우선 설계변수인 제어입력가중행렬  $Q_0$ 을 조정하여 양호한 제어를 얻을 수 있음을 볼 수 있다. 그리고, 제어입력가중행렬  $Q_0$ 의 값이 작아질수록 속응성은 향상되지만 오버슈터가 커짐을 볼 수 있다( $K_p$  증가). 이것은 가 격함수를 사용하는 일반적인 최적제어의 경우(LQ, GPC등)와 동일한 결과라고 볼 수 있다. 감쇠는 1.0인 경우가 가장 크게 나타나는데, 이 경우  $T_d$ 도 역시 가장 큰값을 가짐을 볼 수 있다.

그리고, Fig. 3를 보면 PID이득이 어떤 값으로 수렴하는 특성이 있는데, 수렴값으로 고정된 이득을 가진 PID제어기를 구성하더라도 유사한 결과를 예상할 수 있으므로, 제안된 방법을 고정된 지연시간이 가진 시스템에서 고정된 PID이득을 결정하는데도 이용할 수 있으리라 생각된다.

#### 4. 결론

본 연구에서는 이산시간에서 미지의 시간지연을 가진 시스템에 적용될 수 있는 GPC의 특별한 경우에 대한 제어기 구조를 속도형 PID제어기와 등가 시킴으로써, GPC기법을 이용한 PID이득 조정방법을 구하였다. 그리고, 미지의 파라미터를 가진 시스템에 적용할 수 있도록, 시스템 파라미터를 추정하는 자기동조제어의 기본구조를 채택하여 자기동조 PID제어기를 구성하였다.

그리고 시뮬레이션을 통해서 제안된 알고리즘이 미지의 시간지연을 가진 시스템과 비최소위상 시스템에 대해서도 양호한 성능을 가짐을 보였다. 또한 제안된 알고리즘을 이용하여 고정된 지연시간을 가진 시스템의 PID제어기의 이득을 결정하는데도 이용할수있을 것으로 생각된다.

#### 참고 문헌

- 1] Åström, K.J. et al, "Automatic Tuning of Simple regulators with Specifications on Phase and Amplitude Margins," *Automatica*, Vol.20, No.5, pp.645-651, 1984.
- 2] W. R. Clutt, and E. Goberdhansingh, "Autotuning for Model-Based Predictive Control," *Automatica*, Vol. 26, No. 4, pp. 691-697, 1990.
- 3] F. Cameron, and D.E. Seborg, "A Self-tuning controller with a PID structure," *International Journal of Control*, Vol. 38, pp.401-417, 1983
- 4] P. J. Gawthrop, "Self-tuning PID Controller : Algorithms and Implementation," *Transaction on IEEE Automatic Control*, Vol. 31, No. 3, pp. 201-209, 1986.
- 5] T. Yamamoto, H. Ishihara, S. Omatu and T. Kitamori, "Multivariable Self-Tuning Controller with I-PD Structure," *Proc. of IEEE International Conference on IECON*, pp. 1812-1817, 1991
- 6] D. W. Clarke, C. Mohtadi, and P.S. Tuffs, "Generalized Predictive Control-Part I. The Basic Algorithm," *Automatica*, Vol. 23, pp. 137-148, 1987.
- 7] D. W. Clarke, C. Mohtadi, and P.S. Tuffs, "Generalized Predictive Control-Part II. Extensiona and Interpretation s," *Automatica*, Vol. 23, pp. 149-160, 1987.
- 8] D. W. Clarke, and C. Mohtadi, "Properties of Generalized Predictive Control." *Automatica*, Vol. 25, No. 6, pp.859-875, 1989.
- 9] R. R. Bitmead, M. Gevers, and V. Wertz, *Adaptive Optimal Control*, Prentice Hall, 1990.
- 10] R. Soeterboek, *Predictive Control-A Unified Approach*, Prentice Hall, 1992.
- 11] K. J. Astrom, B. Wittenmark, *Adaptive Control*, Addison-Wesley, 1989.

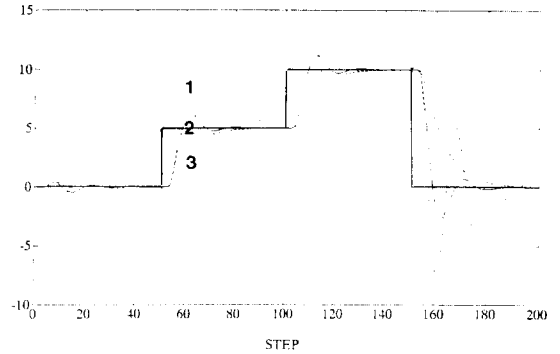


Fig. 1 Output and Reference Input.

(1:  $Q_0=0.5 I_{M \times M}$ , 2:  $Q_0= 1.0 I_{M \times M}$ , 3:  $Q_0= 5.0 I_{M \times M}$ )

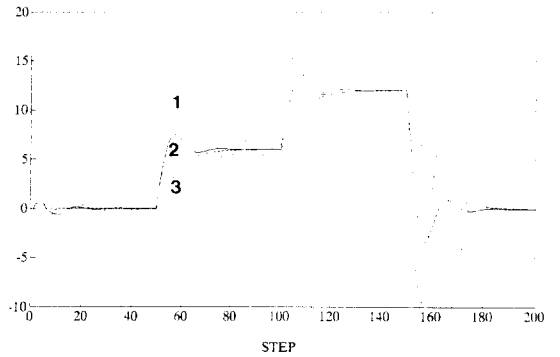


Fig. 2 Control Input.

(1:  $Q_0=0.5 I_{M \times M}$ , 2:  $Q_0= 1.0 I_{M \times M}$ , 3:  $Q_0= 5.0 I_{M \times M}$ )

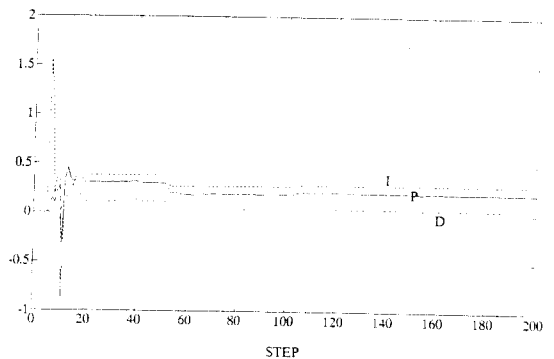


Fig. 3 PID Gains.