

# GPC를 이용한 자기동조 PID 제어기

·유연운 김종만이 창구 김성중  
전북대학교 제어계측공학과

## Self-tuning PID-Controller based on GPC

·Y.W. Yu, J.M. Kim, C.K. Lee, S.J. Kim  
Dept. of Control & Instrumentation Eng.  
Chunbuk National University

### Abstract

The PID controllers which is widely used in the process industry are poorly damped when the dynamic process contains significant dead\_time or when there are random disturbances acting on the plant. GPC is known to be more superior than conventional self\_tuning algorithm in overcoming above problem and prior choice of model order. In this paper, We propose the method which determine the parameter of PID controller from minimization of GPC criterion. The controller has explicit scheme which is comprised of parameter estimation and PID control design. Simulation results show the performance of the proposed self\_tuning PID controller.

### 1. 서론

PID 제어기는 구현의 단순성과 효율성으로 인해 산업현장에서 가장 널리 쓰이고 있다. 그러나 플랜트가 장시간 지연 시간이나 심한 비선형 특성을 갖는다면 제대로 동작하지 못할 뿐 아니라 동특성이 변할 때는 재동조를 해야 하는 번거로움이 있다. 그래서 공정의 입, 출력 정보를 이용하여 온라인으로 최적의 PID 계수값을 결정할 수 있는 방법이 요구된다. [1-2] PID 구조를 갖는 자기 동조 제어기는 Iserman, Gawthrop 등이 연구하였고, 기존의 극배치 방법이 PID 제어기를 동조하기 위해 적용되었으나 적용범위가 협소하다. Banyasz와 Keviczky는 지연시간이 있는 2차 플랜트에서 플랜트 극점을 삭제하고 규정된 위상 마진을 얻기 위해 PID의 다항식  $G(z^{-1})$ 를 사용하였고, [3] Cameron과 Seborg는 GMV(Generalized Minimum Variance) 기법을 사용하여 PID 구조를 갖는 적응 제어기를 제시하였으며 Vega와 Zarop[4]는 이산 시간 제어기 파라미터를 최적화하기 위해 LQG 형태의 성능 지수를 도입하였다.

그러나 적응 제어기를 도입할 때 극 배치 방법은 모델 차수의 가정에 견실하지 못하고 GMV 방법은 지연 시간이 변하면 제대로 동작할 수 없다. [5] 그래서 최대한 가능한 플랜트

의 지연 시간보다 더 큰 시간 영역에 대해 출력을 예측하고 명백한 플랜트 형성을 통해서 지연시간과 모델 차수의 가정에 더욱 견실한 GPC(Generalized Predictive Control) 기법을 이용하여 PID 파라미터를 선정하는 것이 바람직하다.

이 논문에서는 CARIMA 형태의 공정 모델을 사용하였고 최소 자승 오차법으로 추정된 파라미터 이용하여 플랜트의 출력을 예측한다. 예측한 출력과 설정치간의 미래의 제어오차를 고려한 2차의 성능지수를 최소화하여 자기동조 PID 제어기를 설계하였고,  $P(q^{-1})$ 를 이용한 보조출력을 예측하여 모델추종 기능을 갖고 설정치의 변화에 대한 오우버슈트 문제를 개선하도록 알고리즘을 확장하였다.

### 2. PID 제어와 구속 조건

연속 시간에서 표준형 PID 제어기는 (1)식으로 표현된다.

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

$$e(t) = w(t) - y(t) \quad (2)$$

제어기 계수값  $K, T_i, T_d$ 는 각각 이득, 적분 시간, 미분 시간이며,  $u(t)$ 는 시간  $t$ 에서 제어입력을 나타내고  $e(t)$ 는 설정치  $w(t)$ 와 시스템 출력  $y(t)$  사이의 오차를 나타낸다. 샘플링 주기  $T_s$ 로 (1)식을 이산화하고 연속시간과 동일한  $K, T_i, T_d$ 를 사용하여 속도형으로 표현하면 다음과 같다.

$$\Delta u(t) = K [ e(t) - e(t-1) + T_s/T_i e(t) + T_d/T_d (e(t) - 2e(t-1) + e(t-2))] \quad (3)$$

속도형으로 표현된 이산형 PID 제어기를 행렬 형태로 표시하기 위해 제어오차 벡터  $e_t$ 와 제어기 파라미터 벡터  $g$ 를 다음과 같이 정의한다.

$$e_t = [e(t) \ e(t-1) \ e(t-2)] \quad g = [g_0 \ g_1 \ g_2] \\ \Delta u(t) = g_0 e(t) + g_1 e(t-1) + g_2 e(t-2) \quad (4)$$

(3)식과 (4)식을 비교함으로써 연속 시간 계수값과 이산 시간 계수값간의 관계를 표시하면

$$K = -(g_1 + 2g_2) \quad (5a)$$

$$T_d = -T_s g_2 / (g_1 + 2g_2) \quad (5b)$$

$$T_i = -T_s (g_1 + 2g_2) / (g_0 + g_1 + g_2) \quad (5c)$$

이 되며,  $K, T_i, T_d$ 의 계수값들은 유한한 양의 스칼라값이어야 하므로 이산 PID 제어기 파라미터 값은 다음과 같이 구속된다.

$$g_2 > \varepsilon_1 > 0 \quad (6a)$$

$$g_1 + 2g_2 < -\varepsilon_2 < 0 \quad (6b)$$

$$g_0 + g_1 + g_2 > \varepsilon_3 > 0 \quad (6c)$$

단,  $\varepsilon_1, \varepsilon_2, \varepsilon_3$ 는 작은 양수

PID 동작을 보장하기 위해 (6)식의 조건을 만족해야 하며, 그렇지 못한 경우에 최적의 이산 파라미터를 얻기 위해 (6)식을 만족하도록  $g(t)$ 는 투사되어야 한다. 위 식의 부동식은 다음의 행렬 형태로 표현되며,

$$g^T h_i > \varepsilon_i$$

$$\text{단, } h_1 = (0 \ 0 \ 1)^T$$

$$h_2 = (0 \ -1 \ 2)^T$$

$$h_3 = (1 \ 1 \ 1)^T$$

시간  $t$ 에서 최적의 이산 파라미터를 결정하는 과정은 다음의 4단계로 이루어진다.

1.  $\hat{g} = g(t)$
2.  $g^T h_i > \varepsilon_i$  이면 4로 가라.
3.  $\hat{g} = g + \frac{P(t-1) h_i}{h_i^T P(t-1) h_i} (\varepsilon_i - g^T h_i)$
4.  $\hat{g}(t) = g$

앞의 과정에서  $P(t-1)$ 행렬에는 단위 행렬을 사용하므로써  $g(t)$ 의 값에 매우 가까운 최적의 파라미터  $\hat{g}(t)$ 를 결정하였다.

### 3. 자기동조 PID 제어기

공정의 동특성과 잡음 환경을 고려한 미지의 공정모델이 다음과 같은 SISO CARIMA 모델로 표현된다고 가정하자.

$$A(q^{-1}) y(t) = B(q^{-1}) u(t-1) + C(q^{-1}) \varepsilon(t) / \Delta \quad (1)$$

여기서  $y(t)$ 는 시스템 출력,  $u(t)$ 는 제어입력,  $\varepsilon(t)$ 는 평균이 제로이고 분산(variance)이  $\sigma^2$ 인 백색잡음이며,  $A, B, C$ 는 지연 연산자(delay operator)  $q^{-1}$ 의 다항식이고  $\Delta$ 는 차분연산자(differential operator)이다.

$$\begin{aligned} A(q^{-1}) &= 1 + a_1 q^{-1} + \dots + a_n q^{-n} \\ B(q^{-1}) &= b_0 + b_1 q^{-1} + \dots + b_n q^{-n} \\ C(q^{-1}) &= 1 + c_1 q^{-1} + \dots + c_n q^{-n} \end{aligned}$$

공정 모델이 지연 시간을 갖는다면 다항식  $B(q^{-1})$ 의 계수가 첫번째부터 지연시간 만큼 제로가 되고 가변 지연시간일 때는 다항식  $B(q^{-1})$ 의 파라미터에 흡수된다. 유도과정을 간단히 하기 위해  $C(q^{-1}) = 1$ 로 가정하고, (1)식의 모델에 근거하여  $j$ 스텝 미래의 출력을 예측하기 위해서 다음의

diophantine 방정식을 해결해야 한다.

$$1 = E_j(q^{-1}) A \Delta + q^{-j} F_j(q^{-1}) \quad (2)$$

여기에서  $E_j$ 와  $F_j$ 는 주어진  $A(q^{-1})$ 와 출력 예측 구간  $j$ 에 의하여 유일하게 정의되는 다항식이며, (1)식에  $E_j A \Delta$ 를 곱하면

$$E_j A \Delta y(t+j) = E_j B \Delta u(t+j-1) + E_j \varepsilon(t+j)$$

이 되며, (2)식의 관계를 이용하여 정리하면 (3)식과 같은 최적의 출력 예측자를 얻을 수 있다.

$$\begin{aligned} y(t+j) &= G_j \Delta u(t+j-1) + F_j y(t) \\ G(q^{-1}) &= E_j B(q^{-1}) \end{aligned} \quad (3)$$

diophantine 방정식을 순환적(recursive)으로 해결하여 출력 예측자를 계산하는 알고리즘은 표1.2와 같다.

표 1.  $E_j$ 와  $F_j$ 의 계산 알고리즘

1. 초기화  
 $E_1 = 1, F_1 = q(1-A\Delta), \alpha = A\Delta$
2.  $E_j$ 의 계산  
$$e_j^{j+1} = f_0^j$$
  
$$E_{j+1} = E_j + q^{-j} e_j^{j+1}$$
3.  $F_j$ 의 계산  
$$f_{i-1}^{j+1} = f_i^j - f_0^j \alpha_i \quad i \in [1, n_A]$$
  
$$f_0^{j+1} = -f_0^j \alpha_{n_A+1}$$

표 2.  $G_j$ 의 계산 알고리즘

$$G_1 = B$$

$$g_i^{j+1} = g_i^j + f_0^j b_{i-j} \quad i \in [0, n_B+j]$$

단,  $i < j$  이면  $b_{i-j} = 0$ ,

$$j \in [1, N_p-1]$$

자기동조 PID 제어기는 예측한 출력  $\hat{y}(t+j)$ 가 설정치  $w(t+j)$ 를 추종하여 미래의 오차  $e(t+j)$ 를 최소로 하는 제어기 계수값을 결정하도록 설계된다.

PID 파라미터를 각 스텝마다 반복적으로 결정하기 위해 최소화되는 다음과 같은 GPC 사양을 생각하자.

$$J(N_1, N_2) = E \left[ \sum_{j=N_1}^{N_2} [y(t+j) - w(t+j)]^2 + \sum_{j=N_1}^{N_2} \lambda(j) \Delta u(t+j-1)^2 \right] \quad (4)$$

단,  $N_1$ 은 최소 출력구간

$N_2$ 는 최대 출력구간

$\lambda(j)$ 는 제어 가중치 수열

시간  $t+j$ 에서 원하는 설정값  $w(t+j)$ 는 사전에 모든 값이 알겨지는 프로그램 제어를 가정한다. 또한 제어구간  $N_u$ 는 1로 선택하고  $\lambda(j)$ 는  $\lambda$ 로 선택할 때 제어기 파라미터  $g$ 에 관한 함수로 표현되는 GPC 사양은 (5)식으로 된다.

$$J(g) = E \left[ \sum_{j=1}^{N_2} [y(t+j) - w(t+j)]^2 + \lambda \Delta u(t)^2 \right] \quad (5)$$

$$J(g) = \sum_{j=1}^{N_2} [y(t+j) - w(t+j)]^2 + \lambda \Delta u(t)^2 \quad (6)$$

파라미터 추정값  $\hat{\theta}$ 에 의해 계산된 예측자를 사용하면

$$J(g, \theta) = \sum_{j=1}^{N_2} [G_j \Delta u(t+j-1) + F_j y(t) - w(t+j)]^2 + \lambda \Delta u(t)^2 \quad (7)$$

이 되는데 여기서  $j > N_u (=1)$ 인 구간에서  $\Delta u(t+j-1) = 0$ 로 가정하였다. 제어기 파라미터  $g$ 에 관한 식을 유도하기 위해 (7)식에  $G_{j-1} \Delta u(t)$ 를 더하고 맨후 식을 정리하면 비용 함수를 최소화하는  $g$ 는 다음과 같이 된다.

$$J(g) = \sum_{j=1}^{N_2} [y(t+j) - w(t+j) - G_{j-1} \Delta u(t) + G_{j-1} \Delta u(t)]^2 + \lambda \Delta u(t)^2$$

$$g = \min [b_t^T g + 1/2 g^T C_t g] \quad (8)$$

$$g = -b_t / C_t \quad (9)$$

여기서  $b_t$ 와  $C_t$ 는 아래와 같이 주어지고  $\pi_b$ 는 제어 오차를 반영하고  $\pi_c$ 는 제어 신호의 가중치를 반영한다.

$$b_t = \pi_b(t) e_t, \quad C_t = \pi_c(t) e_t e_t^T \quad (10)$$

$$\pi_b(t) = 2 \sum_{j=1}^{N_2} G_{j-1} [y(t+j) - G_{j-1} \Delta u(t) - w(t+j)] \quad (11)$$

$$\pi_c(t) = 2 \left[ \sum_{j=1}^{N_2} G_{j-1}^2 + \lambda \right] \quad (12)$$

이렇게 구한 제어기 파라미터  $g$ 는

$$g = - \frac{\pi_b(t)}{\pi_c(t)} (e_t e_t^T)^{-1} e_t$$

가 되며 역행렬을 구하는 과정을 피하기 위해 역행렬에 관한 명제(matrix inversion lemma)를 사용하면 각 스텝마다 제어기 파라미터를 순환적으로 계산할 수 있다. [8]

$$g(t) = g(t-1) - \frac{\pi_c(t) P(t-1) e_t}{1 + \pi_c(t) e_t^T P(t-1) e_t} [e_t^T g(t-1) + \frac{\pi_b(t)}{\pi_c(t)}] \quad (13)$$

$$P(t) = P(t-1) - \frac{\pi_c(t) P(t-1) e_t e_t^T P(t-1)}{1 + \pi_c(t) e_t^T P(t-1) e_t} \quad (14)$$

구속조건이 없는 경우에 위의 두식은 RLS개선과 아주 흡사한 방식으로 제어기 파라미터  $g$ 는  $e_t^T g(t-1)$ 로 예측된 제어신호와  $\pi_b(t)$ 와  $\pi_c(t)$ 에 의해 반영된 최적의 제어신호 사이의 차이에 의해서 개선된다. RLS와 마찬가지로 이득행렬  $P(t)$ 는 제어오차가 축적되어 무한대가 됨에 따라 제로가 될 것이다.

#### 4. 오우버슈트(overshoot)의 개선

앞장에서 설계된 자기동조 PID 제어기의 건설성과 효율성을 개선하기 위하여 GMV 설계와 같이 보조출력을 예측하고 설정치를 추종하도록 함으로써 알고리즘을 확장하고자 한다. 우선 보조출력  $\Psi(t)$ 를 생각하자.

$$\Psi(t) = P(q^{-1}) y(t)$$

$$P(q^{-1}) = P_n(q^{-1})/P_d(q^{-1})$$

여기서  $P(q^{-1})$ 는 지연연산자  $q^{-1}$ 의 다항식  $P_n(q^{-1})$ 과  $P_d(q^{-1})$ 로 주어진 전달함수이며 정상상태 오차를 제로로 하기 위해  $P(1) = 1$ 이 되도록 선정해야 한다. 그 때 제어기가 초소화하는 비용함수는 다음과 같이 수정된다.

$$J(N_1, N_2) = E \left[ \sum_{j=N_1}^{N_2} [\Psi(t+j) - w(t+j)]^2 + \sum_{j=N_1}^{N_2} \lambda(j) \Delta u(t+j-1)^2 \right] \quad (15)$$

따라서 출력 예측은  $y(t+j)$ 대신에  $\Psi(t+j)$ 를 예측하도록 수정되어야 하며 그 때 diophantine 방정식은 다음과 같이 된다.

$$\frac{P_n}{P_d} = E_j A \Delta + q^{-j} \frac{F_j}{P_d} \quad (16)$$

앞장에서 취급했던 것과 같은 방법으로 하면 예측된 최적의 출력 방정식은 다음과 같이 표현되며, diophantine 방정식을 풀기 위한 절차는 변화가 없고 단지 초기값만이 이동되어진 다.

$$\Psi(t+j) = G_j \Delta u(t+j-1) + F_j y(t) / P_d(q^{-1}) \quad (17)$$

$$G_j = E_j B$$

$$E_1 = \frac{P_n(0)}{P_d(0)}, \quad F_1 = q(P_n - E_1 \alpha), \quad \alpha = A \Delta P_d$$

$$G_1 = \frac{P_n(0)}{P_d(0)} B$$

전달함수  $P(q^{-1})$ 의 역할은 크게 두가지로 분류가 된다. 첫째는 공정제어에서 설정치나 부하의 변동에 의해 출력에 큰 오우버슈트가 발생한다면 이런 오우버슈트 문제를 개선하기 위해  $P(q^{-1})$ 가 사용된다. 둘째는 고성능이 요구되는 제어(high performance control)에 적용할 때  $P(q^{-1})$ 는 근사적인 역퍼루프 모델로 해석된다.  $w(t)$ 의 변화에 대한 페루프 응답은 다음과 같이 주어지며,

$$y(t) = 1/P w(t-k) = M(q^{-1}) w(t-k)$$

$M(q^{-1})$ 는 사용자가 선택한 페루프 모델이다.  $P(q^{-1})$ 를 적절히 사용하여 기준모델 적응제어(MRAC)와 같이 중간 단계 설정치  $w'(t) = M(t)w(t)$ 를 추종하도록 할 수 있으며,  $w'(t)$ 에 대해  $y(t)$ 를 조절하는 최소분산제어(Minimum variance control)를 할 수 있다.

#### 5. 모의실험

제한한 자기동조 PID 제어기의 강인성을 보이기 위해 임의의 공정모델을 대상으로 모의실험을 하였다. 플랜트 파라미터가 일정하고 지연시간을 모르는 경우와 플랜트 파라미터가 삼각파 형태로 변화하는 경우 그리고 지연시간이 변하는 경우를 가정하였는데 설정치에 잘 도달하는 우수한 성능을 보였다. 또한, 플랜트 파라미터가 변화하고 가변지연시간인 경우에는 ziegler\_Nichols 동조규칙을 적용한 고전적인 PID 제어기와 성능을 비교하였는데 제안한 자기동조 제어기는 시변인 경우에 잘 적용하지만 고전적인 PID 제어기는 적응성에 문제가 발생하였다.

### 5.1 다항식 $P(q^{-1})$ 의 효과

공정제어에서 설정치나 부하변동에 따른 오우버슈트를 개선하는  $P(q^{-1})$ 의 효과를 보이기 위해 0 - 200 스텝에서는  $P(q^{-1}) = 1$ 로 선정하였고 200-500 스텝에서는 선정된  $P(q^{-1})$ 를 사용하였다. 그림. 1에서 보는바와 같이  $P(q^{-1})$ 를 이용함으로써 오우버슈트가 현저히 개선되었고 그림. 2의 (a)와 (b)는 각각  $a_1, a_2, b_0, b_1$ 의 추정값을 나타내고 있다. 모든 경우에 있어  $b_0 = 1$ 로 놓고 나머지 파라미터의 초기값은 제로로 가정하고 추정을 하였는데 실제 파라미터에 매우 접근했음을 알 수 있다. 최소자승오차법에서 망각요소(forgetting factor)는 0.98, 공분산 행렬(covariance matrix)은 1000001로 선정하였고  $t = 0$ 인 순간에 0.58의 초기입력을 가하여 파라미터 추정을 하였다. 사용된 공정모델 및 설계변수들은 다음과 같다.

공정모델 :  $1/(1+7.5s)(1+5s)$   
 샘플시간 : 2 sec  
 $N_p$  : 10  
 $\lambda$  : 0  
 $P(q^{-1})$  :  $(1+0.2q^{-1})/1.2$

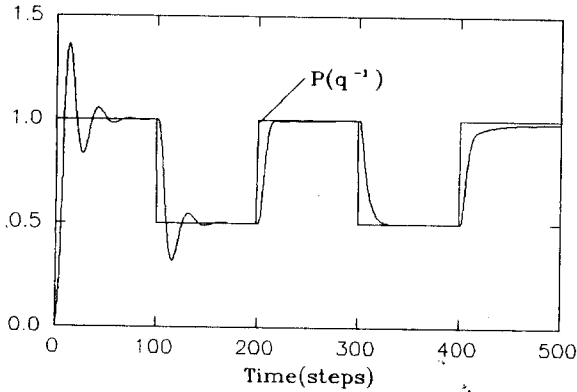


그림 1. 오우버슈트를 개선하기 위한  $P(q^{-1})$ 의 사용  
 FIG. 1. Use of  $P(q^{-1})$  to penalize the overshoot

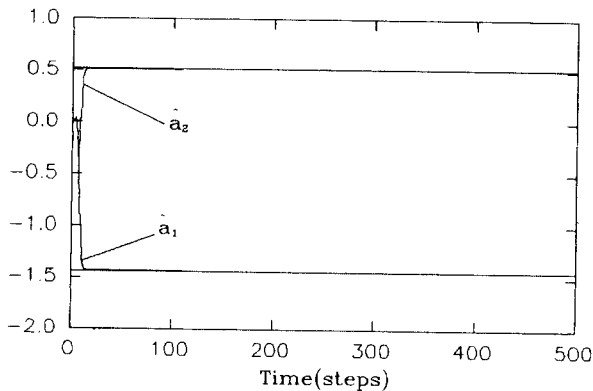


그림 2a. 파라미터  $A(q^{-1})$ 의 추정값  
 FIG. 2a. Estimates of the parameter  $A(q^{-1})$

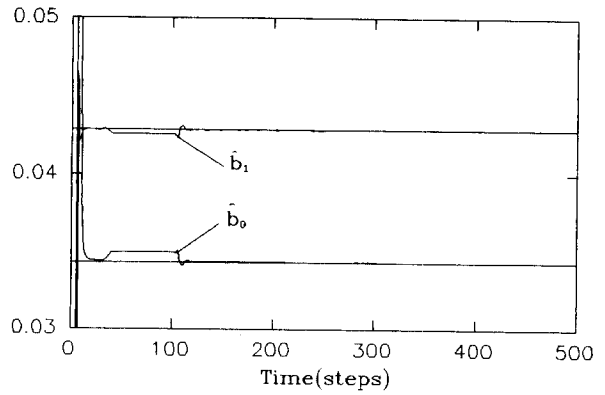


그림 2b. 파라미터  $B(q^{-1})$ 의 추정값  
 FIG. 2b. Estimates of the parameter  $B(q^{-1})$

### 5.2 비최소위상 시스템에서의 성능

공정제어 문제중 하나는 대부분의 플랜트가 비최소위상 특성을 보인다는 것이다. 그래서 제어기가 이러한 시스템에 강인한 성능을 보이는 것이 중요하다. 비최소위상 시스템에서의 성능은 그림 3에서 보여지고 초기의 큰 오우버슈트는 파라미터 추정에서 초기값을 제로로 가정하였기 때문이다. 파라미터 추정이 개선됨에 따라 비최소위상 플랜트도 GMV와는 달리  $\lambda = 0$ 일 때 도 안정하게 제어할 수 있다. 사용된 공정모델 및 설계변수들은 다음과 같다.

공정모델 :  $(1-4s)/(1+4s)(1+10s)$   
 샘플시간 : 4 sec  
 $N_p$  : 15  
 $\lambda$  : 0

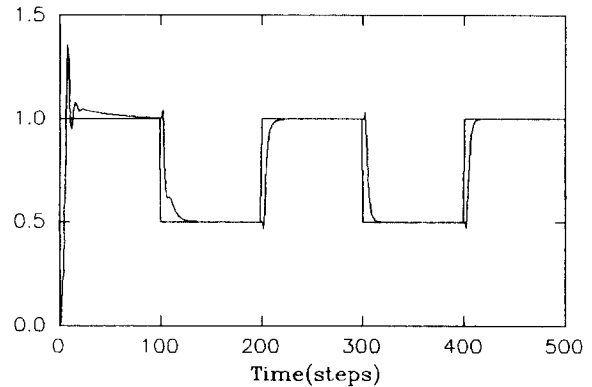


그림 3. 비최소위상 시스템의 출력  
 FIG. 3. Output for the nonminimum-phase system

### 5.3 파라미터 변화가 느린 플랜트

플랜트 파라미터가 변화하는 경우에 대한 제시한 알고리즘의 성능을 살펴보기 위해 파라미터  $a_1, a_2, b_0, b_1$ 을 그림 4와 같이 삼각파 형태로 변화를 주었다. 그림 5에서 보는 바와 같이 자기동조 PID 제어기는 파라미터가 느리게 변화할 때 적응성을 보이지만 고전적인 PID 제어기는 파라미터 변화에 적용하지 못하고 제어 불능 현상을 나타냈다. 모의실험에 사용

한 모델은 과소제동 플랜트를 사용하였고 설계변수의 선정은 아래와 같다.

공정모델 :  $1/(1+10s+40s^2)$   
 샘플시간 : 2 sec  
 $N_p$  : 15  
 $\lambda$  : 0

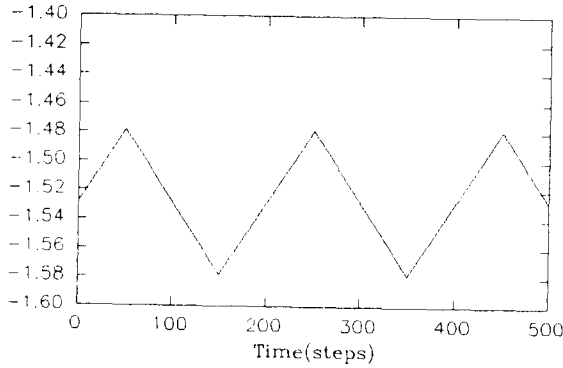


그림 4. 파라미터  $a_1, a_2, b_0, b_1$ 의 변화  
 FIG. 4. Change for the parameters  $a_1, a_2, b_0, b_1$

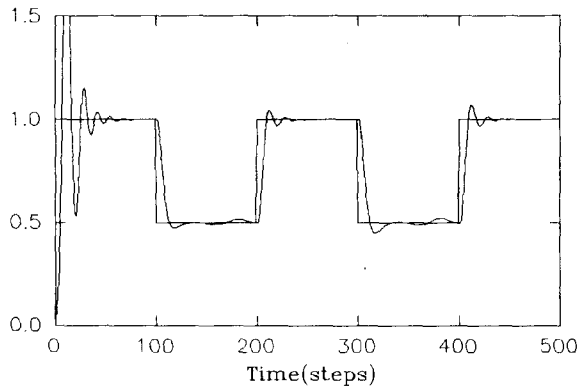


그림 5. 자기동조 PID 제어기의 출력  
 FIG. 5. Output of the proposed self-tuning PID controller

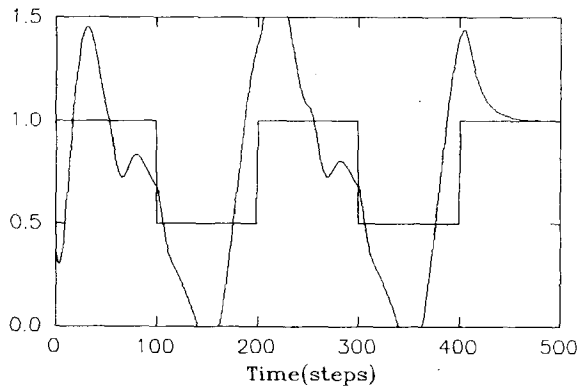


그림 6. Z\_N 동조규칙을 사용한 PID 제어기의 출력  
 FIG. 6. Output of the conventional PID controller using Ziegler\_Nichols rules.

#### 5.4 가변지연시간을 갖는 플랜트

대부분의 자기동조 제어기는 플랜트의 지연시간이 일정하다고 가정하고 있으나 실제 산업현장의 플랜트는 지연시간이 변화하고 있어 가변지연시간에 견실한 성능을 갖는 것이 중요하다. 가변지연시간에 대한 견실성을 보이기 위해 표 2와 같이 지연시간을 변화시키고 자기동조 PID 제어기와 고전적인 PID 제어기의 출력응답을 보였다.

표2. 지연시간의 변화

스텝	지연 시간
0 - 100	0
101 - 200	1
201 - 300	2
301 - 400	1
401 - 500	0

그림 7에서와 같이 자기동조 PID 제어기는 적응성을 보여주고 있으나 그림 8의 고전적인 PID는 지연시간의 변화에 적응하지 못함을 알 수 있다. 과소제동 플랜트를 대상으로 모의 실험하였고 설계변수는 아래와 같다.

공정모델 :  $1/(1+10s+40s^2)$   
 샘플시간 : 2 sec  
 $N_p$  : 10  
 $\lambda$  : 0

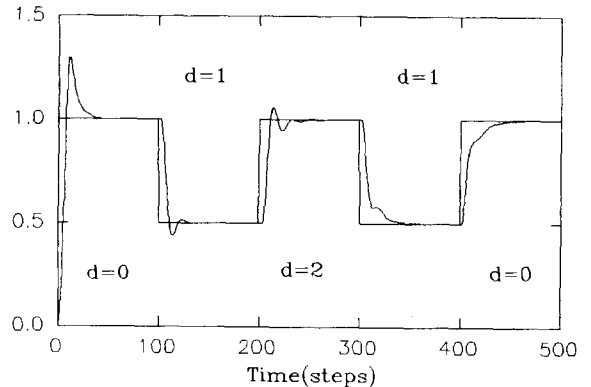


그림 7. 가변지연시간을 갖는 플랜트의 자기동조 PID 제어기의 출력

FIG 7. Output of the self-tuning PID controller for plant with variable delay-time.

#### 6. 결론

공정의 동특성을 CARIMA 모델로 가정하였고, 최소자승수 방법으로 추정된 파라미터를 이용하여 미래의 제어오차를 고려한 2차의 성능지수를 최소화하여 PID 계수값을 결정하고 이를 연속 시간 파라미터에 대응되게 하였다. 그리고 설치치와 부하외란의 변동에 따른 오버슈트를 개선하기 위해  $P(q^{-1})$ 를 사용하여 알고리즘을 확장하였고 여러가지 동특성의 변화를 가정하여 모의실험하였다. 모의실험 결과 PID 제어기의 온라인 동조수단으로 제시한 알고리즘이 많은 플랜트를 동조하기 위해 사용될 수 있고, 플랜트의 지연시간이 모

델 지연시간보다 작다면 가변지연시간을 갖는 플랜트를 안정하게 제어할 수 있을 뿐아니라 파라미터 변화에도 잘 적응함을 보였다.

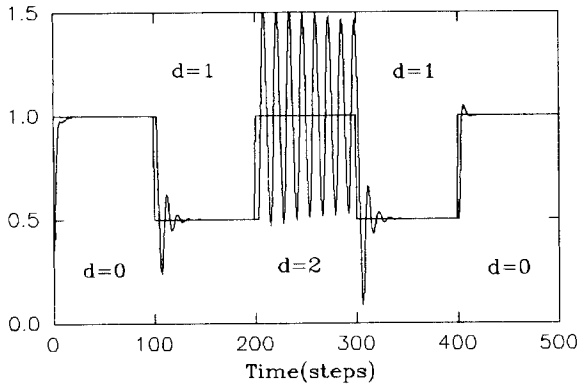


그림 8. 가변지연시간을 갖는 플랜트의 고전적인 PID 제어기의 출력

FIG 8. Output of the conventional PID controller for plant with variable delay-time.

### 참고문헌

- [1] Cameron, F. and D.E. Seborg, "A self-tuning controller with a PID structure", *Int. J. Control*, 38, PP. 401, 1983.
- [2] Isserman, R. and Radke, F., "A parameter-adaptive PID controller with stepwise parameter optimization", *Automatica*, vol. 23, pp. 449-458, 1987.
- [3] P. Vega, C. Prada and V. Aleixandre, "Self-tuning predictive PID controller", *IEE Proc.* Vol. 138, No. 3, May 1991.
- [4] Vega, P. and M.B. Zarrop, "Optimal approaches to self-tuning PID control", *Proc. Adaptive systems in control and signal processing*, Glasgow 1989.
- [5] Clarke, K.W., C. Mohtadi and P.S. Tuffs, "Generalized Predictive Control", *Automatica*, Vol. 23, NO. 2, pp. 137, 1987.
- [6] P.J. Gawthrop, "Self-Tuning PID controller : Algorithms and Implementation", *IEEE Tr. on A.C.*, Vol. 31, No. 3, March 1986.
- [7] Arne Henningsen and Anders Christensen, "A PID Autotuner utilizing GPC and constraint optimization", *Proc. 29th Conference on Dision and Control*, 1980.
- [8] Goodwin, G.C. and K.S. Sin, *Adaptive Filtering, Prediction and control*, Prentice Hall, Englewood Cliffs, NJ, 1984.
- [9] Demircioglu, H. and P.J. Gawthrop, "Continuous-time Generalized Predictive Control (CGPC)", *Automatica*, Vol. 27, NO. 1, PP. 55-74, 1991