

## 쌍일차 공정의 일반형 예측제어

노 균 · 여영구\* · 송형근\* · 윤인섭

서울대학교 화학공학과  
\*한국과학기술연구원 공정제어연구실  
(1990년 11월 6일 접수, 1991년 3월 15일 채택)

## Generalized Predictive Control for Bilinear Processes

Kun Lo, Yeong-Koo Yeo\*, Hyung-Keun Song\* and En Sup Yoon

*Dept. of Chem. Eng., Seoul National Univ.  
\*Process Control Lab., Korea Institute of Science and Technology  
(Received 6 November 1990; accepted 15 March 1991)*

### 요 약

이산시간대 다변수 쌍일차공정에 대해 적분기능을 지닌 새로운 모델인 쌍일차 CARIMA 모델을 제안하고 이를 이용한 일반형 예측제어를 유도하였다. 모델을 이용한 다스텔 최적예측자를 유도하고 이동구간 비용함수를 선정하여 이를 최소화시키는 제어입력을 계산하는 엄밀해법과 근사해법을 제시하였다. 수치모사를 통하여 제안된 제어방법이 공정에 대한 제한된 사전지식만이 반영된 모델을 이용한 경우에도 건실한 제어성능을 발휘함을 확인하였다.

**Abstract**—A generalized predictive controller for discrete-time MIMO bilinear processes is derived based on a new bilinear CARIMA model with integral action. Using this model a multi-step ahead optimal predictor is derived. Two alternative solution methods-rigorous and short-cut-of the minimization problem of a long-range cost-function are introduced and used to calculate control input. Several simulation results show that the proposed control methods have robustness to the limited prior knowledge of the process.

### 1. 서 론

비선형 공정에 대한 제어구조의 합성 또는 제어구조의 해석에 임의의 운전조건 부근에서 선형화된 모델을 이용하는 방법은 보편화되어 있다. 이러한 선형모델을 이용하여 적응제어 및 예측제어의 다양한 기법이 개발되고 있으며 제어이론의 발달과 병행하여 전산제어와 관련된 하드웨어가 급속히 발달되어 운전조건 변화 등의 공정변화에 따라 신속한 모델의 수정 및 보완이 가능하게 되었다. 따라서 제어이론분야에서 선형모델의 적용범위는 날이 갈수록 확장되고 있다. 그러나 선형 제

어이론의 빠른 발달속도에 비추어 실제로 대부분이 비선형인 화학공정의 제어에 이러한 선형 제어기법이 적용되어 성공적인 제어결과를 보인 사례를 찾기가 그리 쉬운 일이 아니다. 이는 공정이 비선형인 경우 선형화된 모델이 이러한 비선형공정의 동특성을 표현하는데 부적합할 뿐만 아니라 조업단계에서의 모델의 수정 및 보완에는 현실적으로 많은 어려움이 존재한다는 사실에 기인한다. 따라서 최근에는 공정상에 존재하는 비선형성을 가능한한 엄밀하게 표현할 수 있는 새로운 모델을 공정의 제어에 이용하는 연구에 대한 관심이 고조되고 있다. 이러한 새로운 모델 가운데 특히 쌍일차(bilinear)

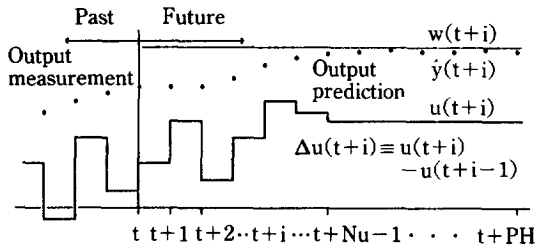


Fig. 1. Prediction horizon(PH) and control horizon(Nu) of the long-range cost-function.

모델은 여타의 비선형 모델에 비하여 수학적인 처리가 용이하며 또한 열전달, 증류, 발열반응 공정 등과 같이 본질적으로 쌍일차 특성을 지닌 공정들을 정확하게 나타내줄 수 있기 때문에 비선형 화학공정의 제어에 응용되는 사례가 점차적으로 증가되고 있다[1-4].

쌍일차 모델을 이용한 제어이론의 발달은 대개 선형 모델을 이용한 기존의 제어이론에 바탕을 두고 있다. 최근에는 선형 모델을 이용한 제어이론들 가운데 그 성능의 우수성이 입증된 방법들을 쌍일차 모델에 확장시켜 적용한 연구결과들이 발표되고 있다.

본 연구에서는 특히 최근들어 제어성능의 우수성과 넓은 적용범위를 지닌 제어방법으로 부각되고 있는 일반형 예측제어(GPC, generalized predictive control) 방법[5-7]에 근거한 쌍일차 모델의 예측제어방법을 유도하였다. 이 방법은 Fig. 1에서와 같이 현재시점  $t$ 를 기준으로 PH개의 예측구간을 설정하고 현재시점 이후의 제어입력  $u$ 의 변화가 공정에 미치는 영향을 모델로부터 예측하여 Nu개의 구간내에서의 제어입력 변화량인  $\Delta u$ 와 공정의 예측출력인  $\hat{y}$ 로 구성된 비용함수를 정의하여 이를 최소화시키는 제어입력을 결정하는 과정으로 다음과 같이 요약할 수 있다.

$$\begin{aligned} \text{Minimize} \\ \Delta u(t) \cdots \Delta u(t+Nu-1) \left\{ \sum_{i=1}^{PH} [\hat{y}(t+i) - w(t+i)]^2 \right. \\ \left. + \sum_{j=1}^{Nu} [\gamma(j) \Delta u(t+j-1)]^2 \right\} \end{aligned} \quad (1)$$

현시점을 기준으로 미리 알고 있는 일정시간 이후의 출력설정치를 반영한 제어입력의 계산을 통해 공정에 가해지는 제어입력의 급격한 변화를 방지할 수 있다는 점이 이 방법의 가장 큰 장점이다. 즉 화학공정과 같이 제어입력부의 기계적인 특성상 짧은 시간내에 급격한 제어입력의 도입이 불가능한 경우, 미래출력의 예측구간을 확장시킴으로써 미래의 공정상태의 변화를 사전에 감지하여 입력조작부의 장기적인 제약에 위배되지 않는 범위내에서 점진적인 입력량의 변화를 공정에 가해줌

으로써 제어목적에 부합되는 제어결과를 얻을 수 있다. 특히 시간지연이 존재하며 공정에 대한 정확한 모델링이 어려운 비선형 화학공정의 제어에 다양한 건실성(robustness)을 지닌 기존의 선형 제어기법에 근거한 쌍일차 예측제어방법을 적용하는 경우 우수한 제어성능을 기대할 수 있다.

본 연구에서는 시간지연의 최소값만을 알고 있는 다변수 공정에 대하여 새로운 형태의 이산시간대 다변수 쌍일차 CARIMA(Controlled Auto-Regressive Integrated Moving-Average) 모델을 제안하고 제안된 모델로부터 다스텔 출력예측자를 유도하여 최적 추적제어에 이용되는 이동구간 비용함수를 최소화시켜 주는 제어입력을 계산하는 쌍일차 일반형 예측제어방법을 유도하였으며 전산모사를 통하여 제시된 제어방법의 유용성을 확인하였다.

## 2. 쌍일차 CARIMA 모델과 출력 예측

제어하고자 하는 다변수의 비선형 화학공정을 나타내어 주기 위한  $n \times n$ 의 다변수 이산시간대 쌍일차 CARIMA(Controlled Auto-Regressive Integrated Moving-Average) 모델은 다음과 같이 쓸 수 있다.

$$\begin{aligned} \Delta(q^{-1})y(t) = & - \sum_{i=1}^{nA} A_i \Delta(q^{-1})y(t-i) \\ & + \sum_{i=0}^{nB} B_i \Delta(q^{-1})u(t-\tau-1-i) \\ & + \sum_{d=0}^D \sum_{j=1}^n \sum_{i=0}^{nC} C_{ij}^d \Delta(q^{-1})[y(t-1-i) \\ & u_j(t-\tau-1-i-d)] \\ & + \sum_{i=0}^{nV} V_i e(t-i) \end{aligned} \quad (2)$$

윗 식에서  $A_i$ ,  $B_i$ ,  $C_{ij}^d$  및  $V_i$ 는  $n \times n$ 의 계수행렬들이다. 또한 지연연산자  $q^{-1}$ 의 다항식  $\Delta(q^{-1})$ 은 차분연산자로서  $(1-q^{-1})I_n$ 으로 정의된다. 이 모델은  $\Delta(q^{-1})$ 이  $n \times n$ 의 단위행렬  $I_n$ 이고  $D$ 가 0인 경우, Svoronos 등[3]이 제안한 쌍일차 모델과 동일해진다.  $nA$ ,  $nB$ ,  $nC$  그리고  $nV$ 는 각각 출력항, 입력항, 출력과 입력의 곱인 쌍일차항 그리고 교란항의 항수를 의미하며 다항식으로 표현하는 경우에는 각각의 다항식 차수를 의미한다. 또한  $\tau$ 는 시간지연의 최소값을 의미하는 상첨자  $jd$ 는  $n$ 개의 입력들 중에서  $j$ 번째 입력항이 최소시간지연  $\tau$ 를 기준으로  $\tau+d$ 만큼의 시간지연으로 쌍일차항에 반영되는 경우를 나타낸다. 출력( $y$ )과 입력( $u$ ) 그리고 교란( $e$ )은 각각  $n \times 1$ 의 벡터로서 다음과 같이 정의된다.

$$y(t) = [y_1(t) \ y_2(t) \ \cdots \ y_n(t)]^T \quad (3)$$

$$u(t) = [u_1(t) \ u_2(t) \ \cdots \ u_n(t)]^T \quad (4)$$

$$e(t) = [e_1(t) \ e_2(t) \ \cdots \ e_n(t)]^T \quad (5)$$

이 때 모델에 포함된 교란  $e(t)$ 는 상호 독립적인 백색 노이즈로서 현재시점  $t$ 를 기준으로 추정가능한 입출력 자료로 구성되는 집합  $F_t$ 에 대하여 다음과 같은 관계를 만족한다고 가정한다.

$$E\{e(t) \mid F_{t-1}\} = 0 \quad (6)$$

$$E\{e(t) \ e(t)^T\} = Q \quad (7)$$

식 (2)의 쌍일차 CARIMA 모델은 공정의 시간지연의 최소값인  $\tau(\geq 0)$ 를 기준으로 공정 출력과 입력의 곱인 쌍일차항에 최소  $\tau$ 에서 최대  $\tau+D$ 까지의 시간지연이 도입되는 경우를 표현할 수 있다. 쌍일차항에서 입력의 시간지연을 변수화함으로써 공정의 모델인식 방법이 공정상의 시간지연 변화를 인식할 수 있는 경우 시간지연이 변화되는 쌍일차 공정에의 적응제어기법 적용이 가능해진다.

쌍일차 모델은 비선형 공정의 물질수지식과 에너지 수지식을 적절한 가정을 도입하여 쌍일차화(bilinearization)시키고 이를 이산화(discretization)시킴으로써 얻어질 수 있다. 증류공정 및 연속회분식 반응기 등의 화학공정을 대상으로 쌍일차 모델을 제시하고 이를 이용한 제어방법 등이 발표되고 있는데, 복잡한 비선형 모델에 비하여 단순하며 수학적인 처리가 용이하다는 점과 선형모델과 비교하여 모델의 정확도가 높다는 점을 장점으로 지적하고 있다[2-4].

다음 식은 식 (2)의 쌍일차 CARIMA 모델을 다항식 행렬로 표현한 것이다.

$$\begin{aligned} A(q^{-1}) \Delta(q^{-1}) y(t) &= B(q^{-1}) \Delta(q^{-1}) u(t-\tau-1) \\ &+ \sum_{d=0}^D \Psi^d(q^{-1}) Y(t-1) \Delta(q^{-1}) \\ &\quad u(t-\tau-1-d) \\ &+ \sum_{d=0}^D \Psi^d(q^{-1}) \hat{Y}(t-1) \\ &\quad u(t-\tau-2-d) \\ &+ V(q^{-1}) e(t) \end{aligned} \quad (8)$$

식 (8)에서 각 항의  $n \times n$  다항식 계수행렬들은 각각 다음과 같이 정의된다.

$$A(q^{-1}) = I_n + A_1 q^{-1} + \cdots + A_{nA} q^{-nA} \quad (9)$$

$$B(q^{-1}) = B_0 + B_1 q^{-1} + \cdots + B_{nB} q^{-nB} \quad (10)$$

$$V(q^{-1}) = I_n + V_1 q^{-1} + \cdots + V_{nV} q^{-nV} \quad (11)$$

$$\Psi^d(q^{-1}) = [C^{1d}(q^{-1}) \mid C^{2d}(q^{-1}) \mid \cdots \mid C^{nd}(q^{-1})] \quad (12)$$

$$C^{jd}(q^{-1}) = C_0^j + C_1^j q^{-1} + \cdots + C_{n_C}^j q^{-n_C} \quad (13)$$

또한 공정의 쌍일차항에 포함된  $Y$ 와  $\hat{Y}$ 는 공정의 출력에 근거한  $n^2 \times n$ 의 행렬로서 다음과 같이 정의된다.

$$Y(t) = \begin{bmatrix} y(t) & 0 & \cdots & 0 \\ 0 & y(t) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & y(t) \end{bmatrix} \quad (14)$$

$$\hat{Y}(t) = \begin{bmatrix} \Delta(q^{-1})y(t) & 0 & \cdots & 0 \\ 0 & \Delta(q^{-1})y(t) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \Delta(q^{-1})y(t) \end{bmatrix} \quad (15)$$

위에서 0은 원소가 모두 0인  $n \times 1$ 의 벡터이다.

식 (8)의 출력  $y$ 에 대하여  $n \times 1$ 의 여과(filtered)출력 벡터  $sf$ 를 다음식과 같이 정의한다.

$$sf(t) = P(q^{-1}) \xi(q) y(t-\tau-1) \quad (16)$$

이 경우 여과기(filter)에 대한 다항식 행렬  $P$ 와 다변수 시스템에서 시간지연을 표시하는 상관자 행렬(interactor matrix)인  $\xi$ 는 각각 다음과 같이 정의된다.

$$P(q^{-1}) = I_n + P_1 q^{-1} + \cdots + P_{nP} q^{-nP} \quad (17)$$

$$\xi(q) = \xi_{\tau+1} q^{\tau+1} \quad (18)$$

식 (8)로 정의된 이산시간대 다변수 쌍일차 CARIMA 모델의 여과출력을 식 (16)과 같이 정의하고 여과출력  $sf$ 에 대하여 다음의 식 (19)로 표현되는 일반형 예측 제어의 이동구간 비용함수를 최소화시켜 주는 제어입력을 유도할 수 있다. 여기서 비용함수는 과거로부터 현재시점  $t$ 까지의 입출력 자료로 구성되는 집합  $F_t$ 에 대한 조건부 기대값(conditional expectation)을 의미한다.

$$\begin{aligned} J(PH, Nu) &= E \left[ \sum_{k=1}^{PH} \| sf(t+\tau+k) - w(t+\tau+k) \|_M^2 \right. \\ &\quad \left. + \sum_{k=1}^{Nu} \| \Delta u(t+k-1) \|_F^2 \mid F_t \right] \end{aligned} \quad (19)$$

위에서  $\|x\|_M^2 = x^T M x$ 으로 정의되며  $n$ 개의 출력에

대한  $n \times 1$ 의 설정치 벡터  $w$ 와  $n$ 개의 입력변화에 대한  $n \times n$ 의 가중치행렬  $r$ 은 다음과 같이 정의된다.

$$w(t) = [w_1(t), w_2(t), \dots, w_n(t)]^T \quad (20)$$

$$r = \text{diag}\{r_1, r_2, \dots, r_n\} \quad (21)$$

또한 PH는 출력예측구간을 표시하며 제어지평(control horizon)  $N_u$ 는 미래의 제어입력변화율이 0이 아닌 구간을 표시한다.

식 (19)의 비용함수를 최소화시켜 주는 제어입력의 계산을 위해서는 현시점  $t$ 를 기준으로 임의의 시간 이후의 여과출력  $sf$ 에 대한 예측자(predictor)를 구성하여야 한다. 입출력 모델을 기준으로 출력에 대한 예측자를 구성하기 위해 Diophantine 방정식을 이용하는 방법은 일반적인 예측제어분야에서 다양하게 이용되고 있다[8]. 식 (16)의 여과출력에 대한 임의의  $k$ -스텝 이후의 예측자를 유도하기 위해 다음과 같은 Diophantine 방정식을 이용한다.

$$P(q^{-1})\xi(q) = F_k(q, q^{-1})A(q^{-1})\Delta(q^{-1}) + q^{\tau+1-k}G_k(q^{-1}) \quad (22)$$

식 (8)의 CARIMA모델의 양변에 다항식 행렬  $F_k(q, q^{-1})$ 와  $q^{k-\tau-1}$ 을 곱하여 정리하고 식 (22)의 관계를 치환하면 Diophantine 방정식의 해인  $n \times n$ 의 다항식 행렬  $F_k$ 와  $G_k$ 가 포함된 다음과 같은 예측자를 구성할 수 있다.

$$\begin{aligned} sf(t+k) &= \hat{sf}(t+k) + \tilde{sf}(t+k) \\ &= P(q^{-1})\xi(q)y(t+k-1-\tau) \end{aligned} \quad (23)$$

식 (23)의 예측자는  $\hat{sf}$ 와  $\tilde{sf}$ 로 구성되는데 이들은 각각 여과출력신호  $sf$ 에 대한 최적출력예측자와 출력예측오차를 의미한다.

$$\begin{aligned} \hat{sf}(t+k) &= \beta_k \Delta u(t+k-2\tau-2) \\ &+ \sum_{d=0}^D \alpha_k^d Y(t+k-2-\tau) \Delta(q^{-1}) \\ &u(t+k-2\tau-2-d) \\ &+ \sum_{d=0}^D \alpha_k^d \tilde{Y}(t+k-2-\tau)u(t+k-2\tau-3-d) + G_k y(t) \end{aligned} \quad (24)$$

$$\tilde{sf}(t+k) = F_k e(t+k-1-\tau) \quad (25)$$

여기서 다항식 행렬인  $\beta_k$ 와  $\alpha_k^d$ 는 다음과 같이 정의한다.

$$\beta_k(q, q^{-1}) \equiv F_k(q, q^{-1}) B(q^{-1}) \quad (26)$$

$$\alpha_k^d(q, q^{-1}) \equiv F_k(q, q^{-1}) \Psi^d(q^{-1}) \quad (27)$$

식 (22)의 Diophantine 방정식에서  $F_k$ 와  $G_k$ 는 다음과 같은 초기값과 순환식을 이용한 단계별 순환(step-recursive)해법[8]을 이용하여 구할 수 있다.

$k$ 가 1인 경우에는

$$G_1(q^{-1}) = q[q^{-\tau-1}P(q^{-1})\xi(q) - \xi_{\tau+1}A(q^{-1})\Delta(q^{-1})] \quad (28)$$

$$\beta_1 = \xi_{\tau+1} B q^{\tau+1} \quad (29)$$

$$\alpha_1^d = \xi_{\tau+1} \Psi^d q^{\tau+1} \quad (30)$$

$$F_1 = F_{\tau+1}^{(1)} q^{\tau+1} = \xi_{\tau+1} q^{\tau+1} \quad (31)$$

$k$ 가 2 이상인 경우에는

$$G_k(q^{-1}) = q[G_{k-1}(q^{-1}) - F_{\tau+2-k}^{(k)} A(q^{-1})\Delta(q^{-1})] \quad (32)$$

$$\beta_k = \beta_{k-1} + G_0^{(k-1)} B q^{\tau+2-k} \quad (33)$$

$$\alpha_k^d = \alpha_{k-1}^d + G_0^{(k-1)} \Psi^d q^{\tau+2-k} \quad (34)$$

$$F_k(q, q^{-1}) = F_{k-1}(q, q^{-1}) + F_{\tau+2-k}^{(k)} q^{\tau+2-k} \quad (35)$$

또한 위에서는 다음의 관계가 성립된다.

$$F_{\tau+2-k}^{(k)} = G_0^{(k-1)} \quad (36)$$

위에서 괄호속의 상첨자( $k$ )는 각각의 계수행렬이  $k$ -스텝 이후의 예측자를 구성하는 다항식 행렬의 구성원소임을 의미한다. 따라서 Diophantine 방정식의 순환해법에서 각각의 다항식 행렬을 다음과 같이 표현할 수 있다.

$$\beta_k(q, q^{-1}) = \sum_{i=-\tau-1}^{nB-\tau-2+k} \beta_i^{(k)} q^{-i} \quad (37)$$

$$\alpha_k^d(q, q^{-1}) = \sum_{i=-\tau-1}^{n\Psi-\tau-2+k} \alpha_i^{(k)d} q^{-i} \quad (38)$$

$$F_k(q, q^{-1}) = \sum_{i=\tau+2-k}^{\tau+1} F_i^{(k)} q^i \quad (39)$$

$$G_k(q^{-1}) = \sum_{i=0}^{nGk} G_i^{(k)} q^{-i} \quad (40)$$

위에서  $nG_k = \max(nP + nV + \tau - k, nA)$ 이다.

### 3. 제어법칙의 유도

식 (19)의 비용함수에서는 앞서 식 (6)과 (7)에서 정의된 교란성분  $e(t)$ 의 성질에 따라 출력예측오차의 기대치가 0이 되기 때문에 비용함수  $J$ 의 최소화과 다음 식 (41)로 정의된 비용함수  $J_1$ 의 최소화의 결과가 동일함을 알 수 있다.

$$J_1(\text{PH}, \text{Nu}) = \left\{ \sum_{k=1}^{\text{PH}} \|\hat{\text{sf}}(t+\tau+k) - w(t+\tau+k)\|_n^2 + \sum_{k=1}^{\text{Nu}} \|\Delta u(t+k-1)\|_r^2 \right\} \quad (41)$$

식 (41)의 비용함수를 벡터와 행렬을 정의하여 다시 쓰면 다음 식 (42)와 같다.

$$J_1(\text{PH}, \text{Nu}) = (\text{SF} - \text{W})^T (\text{SF} - \text{W}) + \tilde{\text{U}}^T \text{R} \tilde{\text{U}} \quad (42)$$

식 (42)에서  $[n \times \text{PH}] \times 1$ 의 벡터 SF와 W,  $[n \times \text{Nu}] \times 1$ 의 벡터  $\tilde{\text{U}}$  그리고  $[n \times \text{Nu}] \times [n \times \text{Nu}]$ 의 행렬 R은 각각 다음과 같이 정의된다.

$$\text{SF} = [\hat{\text{sf}}(t+\tau+1)^T, \hat{\text{sf}}(t+\tau+2)^T, \dots, \hat{\text{sf}}(t+\tau+\text{PH})^T]^T \quad (43)$$

$$\text{W} = [w(t+\tau+1)^T, w(t+\tau+2)^T, \dots, w(t+\tau+\text{PH})^T]^T \quad (44)$$

$$\tilde{\text{U}} = [\Delta u(t)^T, \Delta u(t+1)^T, \dots, \Delta u(t+\text{Nu})^T]^T \quad (45)$$

$$\text{R} = \text{diag}\{r_1, r_2, \dots, r_{\text{Nu}}\} \quad (46)$$

선형 CARIMA 모델을 이용하여 일반형 예측제어기를 유도하는 경우에는 벡터 SF가 벡터  $\tilde{\text{U}}$ 의 함수형태로

표현될 수 있기 때문에 비용함수  $J_1$ 를 최소화하는 제어입력의 변화를 해석적인 방법으로 얻을 수 있다. 그러나 쌍일차 모델을 이용하여 식 (42)의 비용함수를 최소화시켜 주는 제어입력을 결정하는 경우에는 구성된 최적 출력예측자인 SF가 미래의 제어입력과 출력예측치의 함수로 구성되어 있기 때문에 해석적인 방법으로 해를 구할 수가 없다. 따라서 식 (42)의 비용함수를 제어입력의 계산에 이용하기 위해서 식 (24)의 최적 출력예측자를 이용하여 현재점  $t$ 를 기준으로  $\tau + \text{PH}$  이후의 시간까지 여과출력의 최적예측자를 전개하고 다항식 계수행렬을 차수별로 정리하여 벡터와 행렬로 표시하면 다음 식 (47)과 같다.

$$\hat{\text{SF}} = \text{B}\Delta\text{U} + \sum_{d=0}^D \text{A}^d \text{Y}\tilde{\text{U}}^d + \sum_{d=0}^D \text{A}^d \tilde{\text{Y}}\text{U}^d + \text{G} \text{Y}^* \quad (47)$$

식 (47)을 구성하는 각각의 행렬과 벡터는 다음과 같이 정의된다.

$$\hat{\text{SF}} = \begin{bmatrix} \text{SF}_\tau \\ \text{SF} \end{bmatrix} \quad (48)$$

$$\text{SF}_\tau = [\hat{\text{sf}}(t+1)^T, \dots, \hat{\text{sf}}(t+\tau)^T]^T \quad (49)$$

$$\text{SF} = [\hat{\text{sf}}(t+\tau+1)^T, \dots, \hat{\text{sf}}(t+\tau+\text{PH})^T]^T \quad (50)$$

$$\text{B} = \begin{bmatrix} n \times \text{PH} & n \times \tau & n \times n\text{B} \\ \text{O} & \begin{matrix} \beta_{-\tau-1}^{(1)} & \dots & \beta_{n\text{B}-\tau-1}^{(1)} \\ \vdots & & \vdots \\ \beta_{-\tau-1}^{(\tau)} & \dots & \beta_{n\text{B}-2}^{(\tau)} \end{matrix} & \begin{matrix} \beta_{-\tau}^{(1)} & \dots & \beta_{n\text{B}-\tau-1}^{(1)} \\ \vdots & & \vdots \\ \beta_{-1}^{(\tau)} & \dots & \beta_{n\text{B}-2}^{(\tau)} \end{matrix} \\ \text{B}^+ & \text{B}^* & \end{bmatrix} \begin{matrix} n \times \tau \\ n \times \text{PH} \end{matrix} \quad (51)$$

$$\text{B}^+ = \begin{bmatrix} n \times \text{PH} \\ \beta_{-\tau-1}^{(\tau+1)} & \dots & \beta_{n\text{B}-\tau-1}^{(\tau+1)} \\ \vdots & & \vdots \\ \beta_{-\tau-1}^{(\tau+\text{PH})} & \dots & \beta_{n\text{B}-\tau-1}^{(\tau+\text{PH})} \end{bmatrix} n \times \text{PH} \quad (52)$$

$$\text{B}^* = \begin{bmatrix} n \times (\tau + n\text{B}) \\ \beta_{-\tau}^{(\tau+1)} & \dots & \beta_{-1}^{(\tau+1)} & \beta_0^{(\tau+1)} & \dots & \beta_{n\text{B}-1}^{(\tau+1)} \\ \vdots & & \vdots & \vdots & & \vdots \\ \beta_{-\tau+\text{PH}}^{(\tau+\text{PH})} & \dots & \beta_{\text{PH}-2}^{(\tau+\text{PH})} & \beta_{\text{PH}-1}^{(\tau+\text{PH})} & \dots & \beta_{n\text{B}+2 \times \text{PH}}^{(\tau+\text{PH})} \end{bmatrix} n \times \text{PH} \quad (53)$$

$$A^d = \left[ \begin{array}{c|cc} n \times n \times PH & n \times n \times \tau & n \times n \times n\Psi \\ \hline \begin{matrix} \mathbf{O} & & \alpha_{-\tau-1}^{(1)d} & & \alpha_{-\tau}^{(1)d} \dots \alpha_{n\Psi-\tau-1}^{(1)d} \\ & \ddots & \vdots & & \vdots \\ & \alpha_{-\tau-1}^{(\tau)d} \dots \alpha_{-2}^{(\tau)d} & \alpha_{-1}^{(\tau)d} \dots \alpha_{n\Psi-2}^{(\tau)d} \end{matrix} & \begin{matrix} n \times \tau \\ n \times PH \end{matrix} \end{array} \right] \quad (54)$$

$$A^{d+} = \left[ \begin{array}{c} n \times n \times PH \\ \hline \begin{matrix} & & \alpha_{-\tau-1}^{(\tau+1)d} \\ & \ddots & \vdots \\ \alpha_{-\tau-1}^{(\tau+PH)d} \dots \alpha_{-\tau+PH-2}^{(\tau+PH)d} \end{matrix} \end{array} \right] n \times PH \quad (55)$$

$$A^{d*} = \left[ \begin{array}{c} n \times n \times (\tau + n\Psi) \\ \hline \begin{matrix} \alpha_{-\tau}^{(\tau+1)d} \dots \alpha_{-1}^{(\tau+1)d} & \alpha_0^{(\tau+1)d} \dots \alpha_{n\Psi-1}^{(\tau+1)d} \\ \vdots & \vdots \\ \alpha_{-\tau+PH-1}^{(\tau+PH)d} \dots \alpha_{PH-2}^{(\tau+PH)d} & \alpha_{PH-1}^{(\tau+PH)d} \dots \alpha_{n\Psi-2+PH}^{(\tau+PH)d} \end{matrix} \end{array} \right] n \times PH \quad (56)$$

$$\Delta U = \left[ \begin{array}{c} \Delta u(t+PH-1) \\ \vdots \\ \Delta u(t) \\ \hline \Delta U^* \end{array} \right] \begin{matrix} n \times PH \\ n \times (\tau + nB) \end{matrix} \quad (57)$$

$$\Delta U^* = \left[ \begin{array}{c} \Delta u(t-1) \\ \vdots \\ \Delta u(t-\tau) \\ \Delta u(t-\tau-1) \\ \vdots \\ \Delta u(t-\tau-nB) \end{array} \right] n \times (\tau + nB) \quad (58)$$

$$Y\tilde{U}^d = \left[ \begin{array}{c} Y(t+PH+\tau-1) \Delta u(t+PH-d-1) \\ \vdots \\ Y(t+\tau) \Delta u(t-d) \\ \hline Y\tilde{U}^{d*} \end{array} \right] \begin{matrix} n \times n \times PH \\ n \times n \times (\tau + n\Psi) \end{matrix} \quad (59)$$

$$Y\tilde{U}^{d*} = \left[ \begin{array}{c} Y(t+\tau-1) \Delta u(t-d-1) \\ \vdots \\ Y(t) \Delta u(t-d-\tau) \\ Y(t-1) \Delta u(t-d-\tau-1) \\ \vdots \\ Y(t-1-n\Psi) \Delta u(t-d-\tau-n\Psi) \end{array} \right] n \times n \times (\tau + n\Psi) \quad (60)$$

$$\tilde{Y}U^d = \begin{bmatrix} \tilde{Y}(t+PH+\tau-1)u(t+PH-d-2) \\ \vdots \\ \tilde{Y}(t+\tau)u(t-d-1) \end{bmatrix} \begin{matrix} n \times n \times PH \\ n \times n \times (\tau+n\Psi) \end{matrix} \quad (61)$$

$$\tilde{Y}U^{d*} = \begin{bmatrix} \tilde{Y}(t+\tau-1)u(t-d-2) \\ \vdots \\ \tilde{Y}(t)u(t-d-\tau-1) \\ \tilde{Y}(t-1)u(t-d-\tau-2) \\ \vdots \\ \tilde{Y}(t-1-n\Psi)u(t-d-\tau-n\Psi-1) \end{bmatrix} \begin{matrix} n \times n \times (\tau+n\Psi) \end{matrix} \quad (62)$$

$$G = \begin{bmatrix} G_0^{(1)} & \dots & G_{nA}^{(1)} \\ \vdots & & \vdots \\ G_0^{(\tau)} & \dots & G_{nA}^{(\tau)} \end{bmatrix} \begin{matrix} n \times \tau \\ n \times PH \end{matrix} \quad (63)$$

$$G^* = \begin{bmatrix} G_0^{(\tau+1)} & \dots & G_{nA}^{(\tau+1)} \\ \vdots & & \vdots \\ G_0^{(\tau+PH)} & \dots & G_{nA}^{(\tau+PH)} \end{bmatrix} \begin{matrix} n \times PH \end{matrix} \quad (64)$$

$$Y^* = \begin{bmatrix} y(t) \\ y(t-1) \\ y(t-2) \\ \vdots \\ y(t-nA) \end{bmatrix} \begin{matrix} n \times (nA+1) \end{matrix} \quad (65)$$

식 (47)로 정의되는 최적 출력예측자 벡터  $\hat{S}F$  중에서  $t+1$ 에서  $t+\tau$ 의  $\tau$ 개의 구간과  $t+\tau+1$ 에서  $t+\tau+PH$ 의  $PH$ 개의 각각의 구간에 대하여 정의되는  $SF_t$ 와  $SF$ 는 공정의 입출력자료에 대하여 다음과 같은 함수관계를 갖는다.

$$SF_t = fn\{y(t-i), u(t-j); -\tau+1 \leq i \leq n\Psi+1, j \geq 1\} \quad (66)$$

단,  $i \leq -1$ 인 경우  $y(t-i) = \hat{s}f(t-i)$

$$SF = fn\{SF_t, y(t+PH+\tau-i), u(t+PH-i); i \geq -1\} \quad (67)$$

식 (42)의 비용함수를 구성하는 식 (47)로 정의된 최적 출력예측자는 순환식(recursive formula)의 형태를 취하고 있다. 즉 모델을 이용하여 현재시점  $t$ 를 기준으로  $k$ -스텝 이후의 출력예측치를 계산하기 위해서는 현재시점  $t$ 를 기준으로  $(k-1)$ -스텝 이후까지의 출력예측치가 이용되는데 이 경우 출력예측치와 입력의 곱에 의한 쌍일차항의 영향으로 비선형성이 도입되기 때문에 현재시점을 기준으로 하여  $PH$ 개의 여과출력에 대한 예측치가 반영된 비용함수  $J_1$ 을 최소화하는 제어입력의 계산과정은 비선형성을 지닌 순환형 다변수 비용함수에 대한 최적화의 문제가 된다. 이러한 최적화 문제를 해결하기 위해서 본 연구에서는 순환형 비용함수의 원형을

변형시키지 않은 채로 최적화시키는 방법(BGPC)과 순환형 비용함수에 포함된 미래의 출력예측치에 대하여 가정을 도입하여 비용함수를 미래 제어입력만의 함수로 전개하고 선형 일반형 예측제어와 동일한 형태의 제어입력을 계산하는 방법(BGPCA)을 유도하였다.

### 3-1. BGPC(Bilinear Generalized Predictive Control)

BGPC는 순환형 다변수 비용함수를  $n \times n$  다변수 공정에 대하여  $n$ 개의 제어입력이 현재로부터  $N_u$ 개의 구간내에서 변화되는 양을 미리 가정하고 비용함수를 계산하는 과정을 반복함으로써 비용함수를 최소화시켜 주는 입력변화의 최적치를 결정하는 방법으로서 전형적인 다변수 최적화 문제와 동일하다. 이러한 최적화 문제를 해결하는 수치해석방법은 매우 다양하다. 독립변수의 존재 가능 범위에 대한 조건, 즉 제어입력에 대한 제약조건이 존재하는 경우 기존의 선형 일반형 예측제어기는 제어입력의 계산을 QP(Quadratic Programming)에 의존하고 있다[9]. 그러나 쌍일차 공정의 경우에는 비용함수가 순환식(recursive formula)으로 구성되고 쌍일차항의 영향으로 비선형성이 존재할 뿐만 아니라 비용함수의 독립변수에 대한 도함수를 해석적으로 구할 수 없다는 어려움이 있다. 따라서 기존의 다양한 최적화 방법의 이용가능성을 검토한 결과 본 연구에서는 함수형으로 표현이 가능한 비용함수만을 가지고 다변수 최적화의 문제를 해결할 수 있는 방법인 Downhill Simplex법을 이용하였다[10]. 이상과 같은 BGPC를 단계별로 요약하면 다음과 같다.

#### BGPC

단계 1: 현재 시점  $t$ 를 기준으로 입출력자료를 근거로 계산가능한 최적 출력예측자 벡터인  $SF_t$ 를 순차적으로 계산한다.

단계 2: 미래 제어입력 변화에 대한 벡터  $\hat{U}$ 를 다음과 같이 가정한다.

$$\Delta u(t+i-1) = \Delta u_g(t+i-1); i=1, \dots, N_u \quad (68)$$

단계 3:  $k$ 가 1에서 PH까지 PH개의 구간에서 최적 출력예측자  $sf(t+\tau+k)$ 를 계산하여 벡터 SF를 구성한다. 각각의  $k$ 에 대하여 순환형의 형태를 지닌  $sf(t+\tau+k)$ 의 함수관계는 다음과 같이 표현된다.

$$sf(t+\tau+k) = fn\{SF_t, sf(t+\tau+j); 0 \leq j \leq k-1\} \quad (69)$$

단계 4: 단계 3을 통하여 계산된 벡터 SF를 이용하여 비용함수  $J_1$ 를 계산한다. 비용함수가 최소화된 경우 제어입력의 가정치를 제어입력의 최적치로 결정하고 이를 공정에 반영한다. Downhill simplex법에 의한 비용함

수의 최적화에 실패한 경우 새로운 제어입력에 대한 가정치  $\Delta u_g$ 를 선정하여 단계 2의 과정을 반복한다.

### 3-2. BGPCA(Bilinear Generalized Predictive Control A)

선형 CARIMA 모델을 이용한 일반형 예측제어기의 유도과정에서는 모델로부터 유도된 임의의 미래구간내에서 구성된 최적 출력예측자의 구성 성분을 미래의 제어입력 성분과의 관계에 따라 분류하여 다음과 같은 형태로 정리한다.

$$SF = H^0 \hat{U}^0 + F^0 \quad (70)$$

위에서 각각의 행렬과 벡터의 상첨자 0은 Clarke 등[5, 6]이 발표한 GPC 알고리즘에서 표현된 행렬과 벡터를 의미한다.  $\hat{U}^0$ 는 전체 출력예측 PH개의 구간에서 다음과 같이 정의된다.

$$\hat{U}^0 = \begin{bmatrix} \Delta u(t) \\ \Delta u(t+1) \\ \vdots \\ \Delta u(t+PH-1) \end{bmatrix} \quad (71)$$

여기에 제어지평(control horizon)의 개념을 도입하여 전체의 행렬과 벡터의 규모를 축소시키고 다음의 식 (72)와 같은 형태의 최적 출력예측자를 구성하여 식 (42)의 비용함수에 대입한 다음 이를 제어입력에 대해 미분함으로써 식 (73)으로 주어지는 제어법칙을 유도할 수 있다.

$$SF = H_{N_u}^0 \hat{U} + F^0 \quad (72)$$

$$\hat{U} = (H_{N_u}^{0T} H_{N_u}^0 + R)^{-1} H_{N_u}^{0T} (W - F^0) \quad (73)$$

쌍일차 공정에 대하여 쌍일차 CARIMA 모델로부터 유도된 식 (24)의 최적 출력예측자에서 우변의 쌍일차항에 포함되어 순환식으로부터 결정되어야 하는 미래의 출력성분을 이미 알려져 있는 미래의 출력 설정치로 가정하면 선형 일반 예측제어기의 경우와 유사하게 최적 출력예측자를 구성할 수 있다. 즉 다음의 식 (74)와 같은 가정을 도입하면 최적 출력예측자는 다음의 식 (75)와 같이 표현된다.

$$y(t+\tau+i) = w(t+\tau+i), i=1, \dots, PH \quad (74)$$

$$SF = H \hat{U}^0 + F \quad (75)$$

식 (75)를 구성하는  $[n \times PH] \times [n \times PH]$ 의 행렬 H와  $[n \times PH] \times 1$ 의 벡터 F는 각각 다음과 같이 정의된다.

$$H = [B^* + \sum_{d=0}^D A^{d*} Y^d + \sum_{d=0}^D A^{d*} \tilde{Y}^d I_U] I_R \quad (76)$$

$$+ \sum_{d=0}^D A^{d*} \tilde{Y}^d U_{t-1} \quad (77)$$

$$F = B^* \Delta U^* + G^* Y^* + \sum_{d=0}^D A^{d*} Y \tilde{U}^{d*} + \sum_{d=0}^D A^{d*} \tilde{Y} U^{d*}$$

위의 식 (76)과 (77)을 구성하는 각각의 행렬과 벡터 중 일부는 식 (52), (53), (55), (56), (59), (60), (62) 등에서 정의되었으며 나머지 행렬과 벡터들은 다음과 같이 정의된다.

$$Y^d = \begin{bmatrix} n \times d & n \times (PH-d) \\ \hline 0 & 0 \\ \hline 0 & \begin{matrix} Y(t+PH+\tau-1) \\ \vdots \\ Y(t+\tau+d) \end{matrix} \end{bmatrix} \begin{matrix} n \times n \times d \\ n \times n \times (PH-d) \end{matrix} \quad (78)$$

$$\tilde{Y}^d = \begin{bmatrix} n \times (d+1) & n \times (PH-d-1) \\ \hline 0 & 0 \\ \hline 0 & \begin{matrix} \tilde{Y}(t+PH+\tau-1) \\ \vdots \\ \tilde{Y}(t+\tau+d) \end{matrix} \end{bmatrix} \begin{matrix} n \times n \times (d+1) \\ n \times n \times (PH-d-1) \end{matrix} \quad (79)$$

$$I_U = \begin{bmatrix} n \times PH \\ I_n & I_n & I_n & \cdots & I_n \\ & I_n & I_n & \cdots & I_n \\ & & I_n & \cdots & I_n \\ & & & \ddots & \vdots \\ 0 & & & & I_n \end{bmatrix} n \times PH \quad (80)$$

$$I_R = \begin{bmatrix} n \times PH \\ 0 & & I_n & I_n & I_n \\ & I_n & & 0 & \\ & & I_n & & \\ & & & \ddots & \\ I_n & & & & 0 \end{bmatrix} n \times PH \quad (81)$$

$$U_{t-1} = \begin{bmatrix} u(t-1) \\ u(t-1) \\ \vdots \\ u(t-1) \end{bmatrix} n \times PH \quad (82)$$

**Table 1. Model and time-varying plant for SISO simulation**

Model and Processes	Time(min)	A(q <sup>-1</sup> )	B(q <sup>-1</sup> )	C <sup>10</sup> (q <sup>-1</sup> )	τ
Model	0 ≤ t ≤ 400	1 - 0.6q <sup>-1</sup> - 0.3q <sup>-2</sup>	0.8 + 0.5q <sup>-1</sup>	0.3 + 0.2q <sup>-1</sup>	0
Process-I	0 ≤ t ≤ 80	1 - 0.6q <sup>-1</sup> - 0.3q <sup>-2</sup>	0.8 + 0.5q <sup>-1</sup>	0.3 + 0.2q <sup>-1</sup>	0
Process-II	81 ≤ t ≤ 160	1 - 0.6q <sup>-1</sup> - 0.3q <sup>-2</sup>	0.8 + 0.5q <sup>-1</sup>	0.3 + 0.2q <sup>-1</sup>	1
Process-III	161 ≤ t ≤ 240	1 - 0.7q <sup>-1</sup> - 0.21q <sup>-2</sup>	0.91 + 0.4q <sup>-1</sup>	0.25 + 0.12q <sup>-1</sup>	0
Process-IV	241 ≤ t ≤ 320	1 - 0.94q <sup>-1</sup>	0.88	0.23	0
Process-V	321 ≤ t ≤ 400	1 - 0.6q <sup>-1</sup> - 0.3q <sup>-2</sup>	0.5 + 0.8q <sup>-1</sup>	0.3 + 0.2q <sup>-1</sup>	0

앞의 식 (76)에서 정의된 행렬 H로부터 처음 n×N<sub>Nu</sub>개의 열로 구성되는 [n×PH]×[n×N<sub>Nu</sub>]의 행렬 H<sub>Nu</sub>를 구성하여 다음과 같은 제어법칙을 유도할 수 있다.

$$\tilde{U}(H_{N_u}^T H_{N_u} + R) H_{N_u}^T (W - F) \quad (83)$$

위의 제어법칙은 최적 출력예측자의 구성에 있어서 쌍일차항에 포함되는 미래 출력예측치가 이미 알고 있는 미래의 출력설정치와 동일하다는 가정을 근거로 하고 있으며 모델의 쌍일차항을 구성하는 계수행렬이 반영되어 있다. 이러한 가정은 선형 일반형 예측제어에 이용되었던 CARIMA 모델이 지닌 적분기능의 영향으로 출력과 설정치간의 편차(offset)를 제거하는 작동 경향이 쌍일차 CARIMA 모델의 경우에도 유사하게 작동될 것임을 감안한 것이다. 다스텔의 예측자를 이용한 제어방법에 있어서는 이러한 가정으로 발생하는 예측오차가 제어 결과에 미치는 영향이 극소화될 것을 기대할 수 있으며 또한 미래 출력예측자를 설정치 부근에서 선형화시킨 모델로부터 구성하는 경우와 유사한 결과를 얻게 되므로 그 가정은 매우 타당한 것이라고 볼 수 있다.

## 4. 모사 결과

다음은 쌍일차 CARIMA 모델로부터 유도된 두 가지의 일반형 예측제어방법인 BGPC와 BGPCA의 제어 성능에 대한 모사결과이다. 쌍일차 공정에 대한 다스텔 예측제어 방법의 우수성을 입증하기 위하여 제어기의 튜닝과 관련된 입력가중치 행렬(R), 제어지평(Nu) 및 출력예측구간(PH) 등의 파라미터들 중에서 R은 전체 구성원소가 0인 행렬로 Nu는 1로 고정시키고 출력예측구간만을 변화시키며 모사하였다.

우선적으로 공정의 동특성이 시간에 따라 다양하게 변화되는 가상의 SISO 공정에 대하여 제안된 제어방법을 적용함으로써 제어기의 건실성(robustness)을 확인하고, 다음으로 MIMO 공정을 대상으로 기존의 선형모델을 이용한 GPC와 본 연구에서 유도한 BGPC 및

BGPCA를 적용하여 각각의 제어결과를 비교함으로써 제안된 제어방법의 우수성을 입증하고자 한다.

모사에 이용될 n×n의 비선형 공정과 이의 모델은 다음의 식 (84)와 같은 이산시간대 쌍일차 CARIMA 모델로 표현된다.

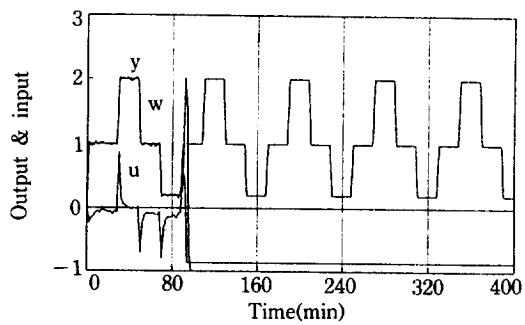
$$\begin{aligned} A(q^{-1}) \Delta(q^{-1})y(t) &= B(q^{-1})\Delta(q^{-1})u(t-1-\tau) \\ &+ \sum_{d=0}^D \sum_{j=1}^n C^{jd}(q^{-1})\Delta(q^{-1}) \\ &[y(t-1)u_j(t-1-\tau-d)] \\ &+ V(q^{-1})e(t) \end{aligned} \quad (84)$$

공정의 출력은 여과기(filter) 없이 측정하는 경우로 고정하였고 공정의 출력측정에는 교란이 도입되도록 하여 계단과 형태의 출력설정치에 대한 추적성능 및 출력측정에 도입되는 교란성분의 제거성능을 모사하였다. 또한 모든 경우에 있어서 샘플링간격은 1분으로 가정하고, 교란성분에 대한 다항식 계수행렬인 V(q<sup>-1</sup>)는 I<sub>n</sub>으로 가정하였다.

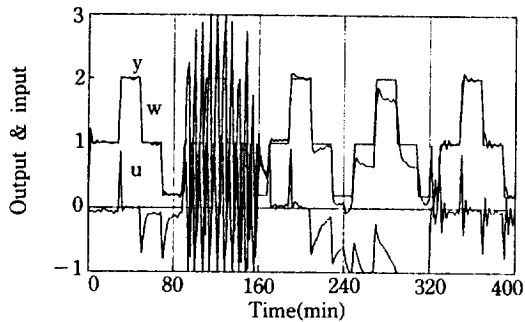
### 4-1. SISO(Single-Input Single-Output) 공정

제어에 이용되는 모델과 공정의 시간에 따른 동특성 변화는 Table 1에 요약된 바와 같다. 고정된 모델에 대하여 모델과 공정이 완전히 일치되는 공정-I, 시간지연이 1회의 샘플링간격 만큼 차이가 있는 공정-II, 파라미터가 일치되지 않는 공정-III, 차수가 다른 공정-IV, 비최소 위상(non-minimum phase)시스템인 공정-V 등으로 공정의 변화는 매우 다양하다. 공정의 출력측정에 도입되는 교란성분은 0.2+0.05v(t)로 이 때 v(t)는 평균 값이 0인 정규 백색노이즈(normalized white noise)이다. 또한 쌍일차항의 시간지연은 균일한 경우로 식 (84)에서 D는 0이다.

이상과 같은 쌍일차 SISO 공정에 대하여 제안된 BGPC 및 BGPCA 제어방법을 적용한 결과는 다음과 같다. Fig. 2, 3, 4 및 5는 출력예측구간(PH)를 각각 1, 2, 7 그리고 8로 변화시킨 제어결과이다. 각각의 그림에서 (a)는 BGPC를 그리고 (b)는 BGPCA를 적용한

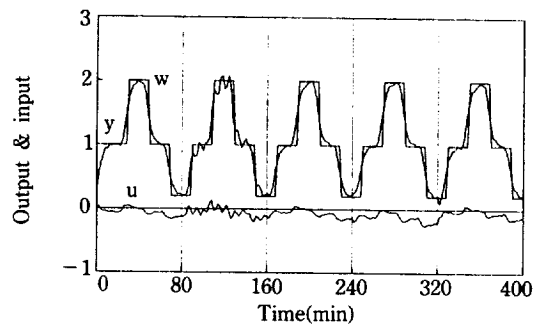


(a) BGPC result

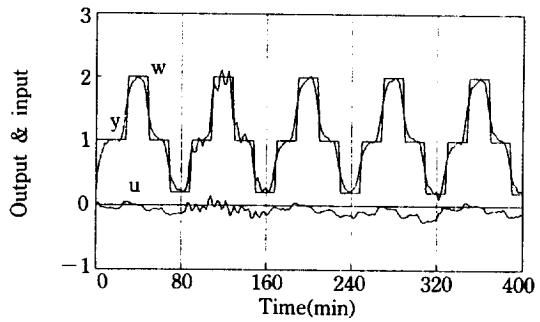


(b) BGPCA result

Fig. 2. BGPC and BGPCA for time-varying SISO plant. (PH=1, Nu=1, R=0)

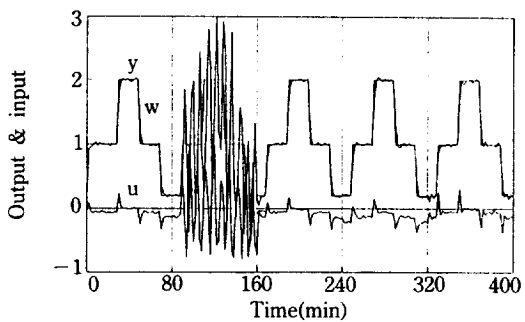


(a) BGPC result

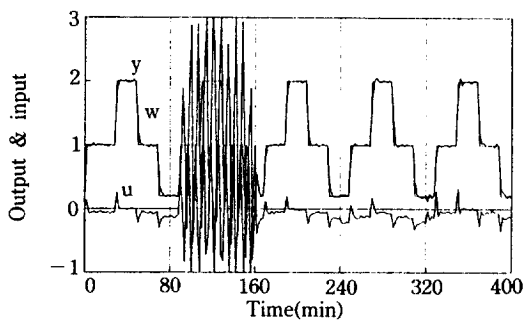


(b) BGPCA result

Fig. 4. BGPC and BGPCA for time-varying SISO plant. (PH=7, Nu=1, R=0)

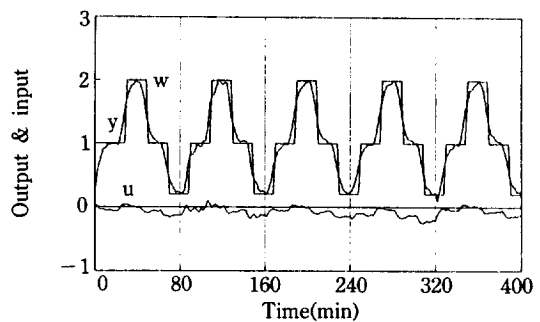


(a) BGPC result

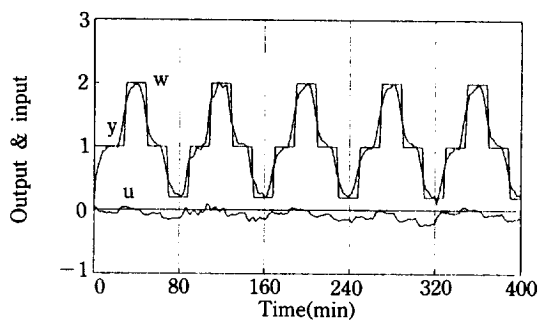


(b) BGPCA result

Fig. 3. BGPC and BGPCA for time-varying SISO plant. (PH=2, Nu=1, R=0)



(a) BGPC result



(b) BGPCA result

Fig. 5. BGPC and BGPCA for time-varying SISO plant. (PH=8, Nu=1, R=0)

출력(y), 설정치(w) 그리고 제어입력(u)를 표시한 것이다. Fig. 2의 결과를 살펴보면 쌍일차 공정에 대한 다스텔 예측제어방법의 필요성을 알 수 있다. 쌍일차 모델을 이용하는 경우 현재시점을 기준으로 바로 다음 스텝의 출력은 쉽게 예측될 수 있다. 즉 현재까지의 제어입력 및 출력을 이용하면 BGPCA의 유도과정에서 도입된 것과 같은 가정이 없어도 단스텔 예측자를 유도할 수 있다. 그러나 단스텔 예측자가 이용되는 경우에는 Fig. 2에서 볼 수 있듯이 모델과 공정이 완벽하게 일치되는 공정-I의 경우를 제외하고는 전체적인 제어 결과가 만족스럽지 않게 나타난다. 따라서 단스텔 예측자를 이용한 기존의 예측제어방법들[11, 12]을 이용하는 경우에는 공정이 변화되는 각각의 구간에 대하여 제어를 새롭게 튜닝해야만이 만족할 만한 제어결과를 기대할 수 있다. 출력예측구간의 변화에 따른 제어결과를 살펴보면 모델과 공정이 완벽하게 일치되는 공정-I에 대해서는 PH에 관계없이 안정한 제어결과를 보이며 PH의 증가에 따라 교란성분의 영향이 점차적으로 완화되고 출력과 제어입력이 완만한 변화를 보인다. 공정-II는 모델과 공정의 시간지연이 다른 경우로 PH의 증가가 안정된 제어를 위한 필수적인 요소임을 확인할 수 있다. 모델과 공정의 파라미터가 다른 공정-III과 추수가 다른 공정-IV의 경우에 대해서도 PH가 2 이상이면 안정된 제어성능이 발휘됨을 알 수 있다. 공정이 비최소

위상 시스템인 공정-V의 경우에는 PH의 증가에 따라 제어입력의 변화의 폭이 현저하게 감소되는 결과를 보인다.

Table 1과 같이 다양하게 변화되는 쌍일차 SISO 공정에 대하여 고정된 모델을 이용한 단순예측제어방법인 BGPC와 BGPCA의 제어결과로부터 쌍일차 공정에 대하여 제안된 다스텔 예측제어방법이 매우 건실한 제어 성능을 지니고 있음을 확인할 수 있다. 또한 제안된 두 가지의 제어방법의 작동추세가 전반적으로 유사하게 나타난다. 이러한 사실로부터 BGPCA의 유도과정에서 도입된 가정이 타당함을 알 수 있다.

한편 모델 인식방법을 적용한 적응예측제어방법도 이용할 수 있다. 모델 인식방법인 방정식 오차법이 결합된 BGPCA 제어방법은 공정의 동특성이 심하게 변화되는 공정에 대하여 만족할 만한 제어결과를 보인다 [13].

#### 4-2. MIMO(Multi-Input Multi-Output) 공정

다음은 제안된 BGPC 및 BGPCA 제어방법을 쌍일차 MIMO 공정의 제어에 적용하고 그 제어결과를 기존의 선형모델을 이용한 GPC의 제어결과와 비교한 것이다. 제어에 이용되는 모델과 공정의 시간에 따른 동특성 변화를 표현하기 위하여 다음과 같은 10개의 다항식 계수행렬을 정의하였다.

$$M_1 = \begin{bmatrix} 1+0.2q^{-1}-0.3q^{-2} & -0.4q^{-1}+0.2q^{-2} \\ 0.5q^{-1}-0.7q^{-2} & 1-0.3q^{-1}-0.4q^{-2} \end{bmatrix} \quad (85)$$

$$M_2 = \begin{bmatrix} 2+4q^{-1} & q^{-2}(0.5+q^{-1}) \\ q^{-1}(2-1.4q^{-1}) & 1.5q^{-4} \end{bmatrix} \quad (86)$$

$$M_3 = \begin{bmatrix} 0.5+0.62q^{-1} & q^{-1}(0.02+0.02q^{-1}+0.1q^{-2}) \\ q^{-1}(0.1-0.05q^{-1}) & q^{-3}(1.2+0.3q^{-1}) \end{bmatrix} \quad (87)$$

$$M_4 = \begin{bmatrix} 0.2+0.3q^{-1} & q^{-2}(0.1+0.2q^{-1}) \\ q^{-1}(0.2+0.1q^{-1}) & q^{-3}(0.6+0.2q^{-1}) \end{bmatrix} \quad (88)$$

$$M_5 = \begin{bmatrix} 1+0.16q^{-1}-0.28q^{-2} & -0.35q^{-1}+0.15q^{-2} \\ 0.4q^{-1}-0.66q^{-2} & 1-0.2q^{-1}+0.42q^{-2} \end{bmatrix} \quad (89)$$

$$M_6 = \begin{bmatrix} 1.2+4.4q^{-1} & q^{-2}(0.55+1.2q^{-1}) \\ q^{-1}(2.2-1.3q^{-1}) & 1.4q^{-4} \end{bmatrix} \quad (90)$$

$$M_7 = \begin{bmatrix} 0.45+0.6q^{-1} & q^{-1}(0.01+0.02q^{-1}+0.1q^{-2}) \\ q^{-1}(0.13-0.045q^{-1}) & q^{-3}(1.1+0.32q^{-1}) \end{bmatrix} \quad (91)$$

$$M_8 = \begin{bmatrix} 0.22+0.31q^{-1} & q^{-2}(0.12+0.2q^{-1}) \\ q^{-1}(0.18+0.11q^{-1}) & q^{-3}(0.5+0.21q^{-1}) \end{bmatrix} \quad (92)$$

Table 2. Model and time-varying plant for MIMO simulation

Model and Processes	Time(min)	A	B	C <sup>10</sup>	C <sup>21</sup>	C <sup>11</sup>	C <sup>21</sup>	Square error sum of parameters
Model	$0 \leq t \leq 450$	M <sub>1</sub>	M <sub>2</sub>	M <sub>3</sub>	M <sub>4</sub>			
Process-I	$0 \leq t \leq 150$	M <sub>5</sub>	M <sub>6</sub>	M <sub>7</sub>	M <sub>8</sub>			0.957
Process-II	$151 \leq t \leq 300$	M <sub>5</sub>	M <sub>6</sub>	M <sub>3</sub>	M <sub>4</sub>	M <sub>9</sub>	M <sub>10</sub>	2.756
Process-III	$301 \leq t \leq 450$	M <sub>1</sub>	M <sub>2</sub>			M <sub>9</sub>	M <sub>10</sub>	4.642

$$M_9 = \begin{bmatrix} 0.2 + 0.5q^{-1} & q^{-1}(0.01 + 0.01q^{-1} + 0.05q^{-2}) \\ q^{-1}(-0.1 - 0.04q^{-1}) & q^{-3}(1 + 0.1q^{-1}) \end{bmatrix} \quad (93)$$

$$M_{10} = \begin{bmatrix} 0.1 + 0.1q^{-1} & q^{-2}(-0.1 + 0.3q^{-1}) \\ q^{-1}(-0.2 - 0.1q^{-1}) & q^{-3}(0.3 + 0.5q^{-1}) \end{bmatrix} \quad (94)$$

BGPC 및 BGPCA의 적용에는 위에서 정의된 다항식 계수행렬을 이용한 Table 2와 같은 모델과 공정을 이용하였다. 공정-I은 모델과 계수행렬이 다른 경우이고, 공정-II와 공정-III은 식 (84)에서 D가 1인 경우로 공정-II는 쌍일차항에 복합적인 시간지연이 나타나고 공정-III은 쌍일차항의 시간지연이 모델과 1회의 샘플링 간격 만큼 차이를 나타낸다. 제어대상공정의 난이도를 정량적으로 분석하기 위하여 모델과 공정의 계수행렬을 구성하는 계수들의 오차를 제공하여 합산한 값을 Table 2에 표시하였다.

선형모델을 이용한 GPC를 쌍일차 공정에 적용하여 쌍일차 모델을 이용한 제어방법과 그 제어성능을 비교하는 조건을 최대한으로 접근시키기 위하여 GPC는 최소자승법을 이용하여 매회 샘플링마다 모델의 파라미터를 수정하도록 하였고 다항식 계수행렬인  $A(q^{-1})$ 와  $B(q^{-1})$ 의 초기값을 각각  $M_1$ 과  $M_2$ 로 하였다. 또한 다항식 계수행렬의 차수  $n_A$ 와  $n_B$ 가 각각 2와 4인 경우 및 3과 5인 경우 두 가지를 비교의 대상으로 삼아 쌍일차항의 영향이 선형모델에 반영될 수 있도록 하였다. 모든 경우에 있어서 공정과 모델의 최소 시간지연  $\tau$ 는 0이며 PH는 10으로 고정시켰고 교란성분은 다음과 같다.

$$e(t) = \begin{bmatrix} 0.3 + 0.3v_1(t) \\ 0.15 + 0.015v_2(t) \end{bmatrix} \quad (95)$$

$v_i(t)$ : normalized zero-mean white noise

이상과 같이 시간에 따라 상태변화가 발생하는 비최소위상을 지닌 다변수 쌍일차 공정에 대한 제어모사의 결과는 다음과 같다. Fig. 6과 7은 각각 BGPC와 BGPCA를 이용한 제어결과이다. 그리고 Fig. 8과 9는 GPC를 이용한 제어결과를 나타낸다. Fig. 8은  $n_A$ 가 2 그리고  $n_B$ 가 4인 선형모델을 이용한 경우이고, Fig. 9는  $n_A$

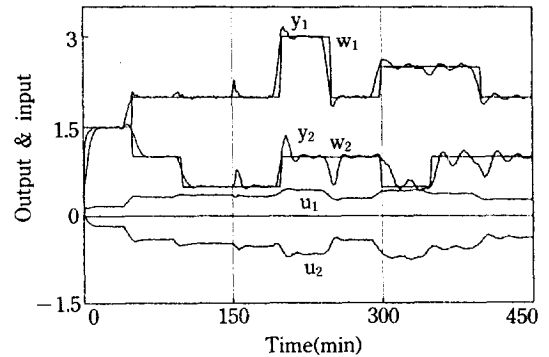


Fig. 6. BGPC for time-varying MIMO plant.  
(PH=10, Nu=1, R=0)

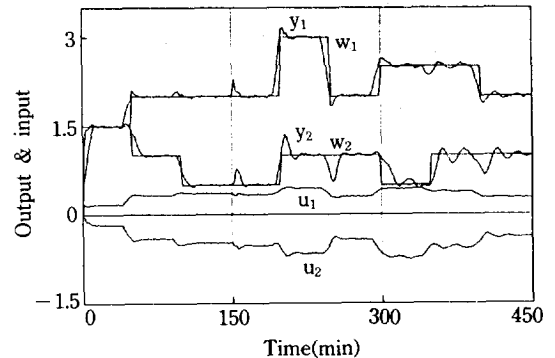


Fig. 7. BGPCA for time-varying MIMO plant.  
(PH=10, Nu=1, R=0)

3  $n_B$ 가 5인 경우이다. PH를 달리한 제어결과와 모사에 있어서는 SISO 공정의 제어결과와 유사한 결과를 관찰할 수 있었다. MIMO 공정에 있어에서도 제안된 BGPC와 BGPCA 작동추세가 유사하게 나타났으며 선

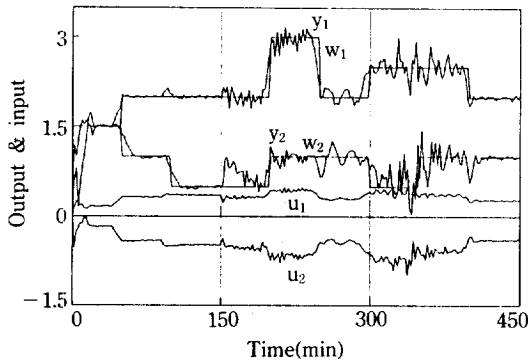


Fig. 8. Adaptive GPC for time-varying MIMO plant.  
(PH=10, Nu=1, R=0, nA=2, nB=4)

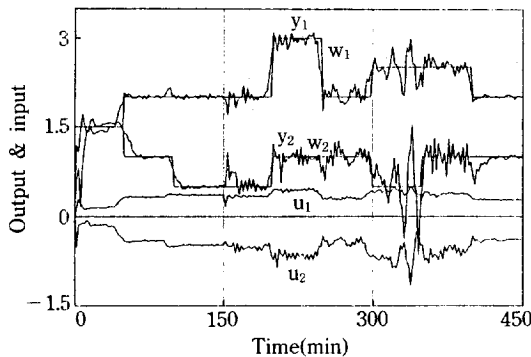


Fig. 9. Adaptive GPC for time-varying MIMO plant.  
(PH=10, Nu=1, R=0, nA=3, nB=5)

형모델을 이용한 경우에 비교하여 우수한 제어성능이 발휘됨을 확인할 수 있다. 공정-I을 살펴보면 공정의 쌍일차항을 구성하는 계수들의 크기가 상대적으로 작음을 알 수 있다. 따라서 Fig. 8과 9에서 공정-I에 대한 제어결과는 쌍일차 모델이 이용된 경우와 비교하여 큰 차이를 보이지 않는다. 그러나 쌍일차항의 영향이 복합적인 시간지연으로 공정-I에 비하여 크게 나타나는 공정-II의 경우와 쌍일차항의 시간지연이 모델과 1회의 샘플링간격 만큼 차이를 나타내는 공정-III의 경우에 있어서는 적응예측제어기법인 GPC의 제어성능이 본 연구에서 제안된 단순예측제어방법들의 제어성능에 미치지 못함을 관찰할 수 있다.

따라서 비선형성이 심하고 특히 이러한 비선형성에 대한 쌍일차항의 모델링이 가능한 공정에 대해서는 제안된 제어방법이 기존의 선형모델을 이용한 제어방법에 비하여 우수한 제어성능을 지닌다. 그리고 쌍일차 모델의 인식방법에 대하여 발표된 기존의 연구결과들[14, 15]을 바탕으로 다변수 쌍일차 공정모델의 인식방법을

개발하고 이를 이용한 적응예측제어방법을 체계화하는 경우 제안된 제어방법의 제어성능은 더욱 향상될 것으로 예상된다.

SISO 공정과 MIMO 공정의 제어에 있어서 동일한 출력예측구간이 설정된 경우 BGPC 방법은 오차허용범위에 따라 최적화 방법으로 이용된 downhill simplex법의 내부 반복 계산회수가 일반적으로 50회 내지 60회 정도에 달한다. 따라서 출력예측구간이 크고 비용함수를 최소화하는 정확한 제어입력의 최적치를 계산하는 경우에는 현재의 전산기 계산속도와 적용된 최적화 방법의 성능상 실시간 조업에 BGPC 방법이 이용되는 것은 샘플링 간격이 충분히 큰 공정의 경우를 제외하고는 불가능하다. 모사결과에서 알 수 있듯이 미래출력에 대한 가정이 도입된 BGPCA를 이용한 경우의 제어결과가 BGPC의 경우와 전반적으로 유사하게 나타나므로 샘플링 시간간격이 짧은 공정의 실시간 조업에는 BGPCA 방법을 사용하는 것이 바람직하다.

## 5. 결 론

본 연구에서는 다변수 쌍일차 공정에 대한 새로운 형태의 모델식으로서 다변수 이산시간대 쌍일차 CA-RIMA 모델을 제안하였으며 이를 바탕으로 쌍일차 다변수 공정에 대한 일반형 예측제어방법을 유도하였다. 제안된 모델은 쌍일차항에 최소  $\tau$ 에서 최대  $\tau+D$ 의 시간지연의 변화가 일어나는 경우를 표현할 수 있다. 본 연구에서 제안된 BGPC와 BGPCA 제어방법은 제안된 모델을 이용하여 여과출력에 대한 다스텔 최적 출력예측자를 유도함으로써 미래의 출력예측치와 설정치간의 오차와 제어입력의 변화가 반영된 비용함수를 최소화시켜 주는 제어입력을 계산한다. 비선형성을 지닌 순환형 비용함수를 downhill simplex법을 이용하여 최적화하는 BGPC와 최적 출력예측자에 포함된 미래 출력치를 이미 알고 있는 출력 설정치로 가정하여 비용함수를 최소화하는 해석적인 최적 제어입력을 계산하는 BGPCA가 서로 유사한 제어성능을 발휘함을 전산 모사결과로부터 확인할 수 있었다. 제안된 두 가지 방법은 공정에 대한 부분적인 사전지식만으로도 견실한 제어성능이 발휘되며 모델에 포함된 적분기능의 작용으로 공정의 출력치가 설정치에 편차없이 수렴하는 결과를 보이고 있다. 또한 다스텔 예측제어방법의 일반적인 성질로 출력예측구간의 확장에 따라 급격한 제어입력의 변화를 억제하는 효과를 발휘함으로써 입력조작부의 장치적인 제약이 존재하는 화학공정에 대한 적용가능성이 매우 높음을 확인하였다. 기존의 선형모델을 이용한 GPC와 제어성능을 비교한 결과, 제안된 제어방

법들이 공정에 쌍일차특성이 존재하는 경우에 대하여 더 우수한 제어성능을 발휘함을 확인하였다.

앞으로 계속 연구되어야 할 사항은 제안된 제어방법의 안정성을 증명하고 다양한 공정을 대상으로 제어성능을 모사하고 실제 비선형 공정의 제어에 적용하여 제어기 튜닝 파라미터인 입력가중치 행렬(R), 출력예측구간(PH) 그리고 제어지평(Nu)이 제어성능에 미치는 영향을 분석하여 공정특성에 따른 제어기튜닝에 대한 지침을 마련하는 것이다. 또한 BGPC에 다양한 최적화 방법을 적용하여 좀더 효율적으로 최적 제어입력을 계산하는 방법 및 다변수 공정의 적응예측제어방법과 관련된 연구도 수반되어야 한다.

## NOMENCLATURE

$A_i, B_i, C_i^d, V_i$	: model parameter matrix
$A, B, C^d, V$	: model parameter matrix
$A^d, B, G$	: optimal predictor coefficient matrix
$d$	: time delay variation
$D$	: upper bound of time delay variation
$e$	: disturbance vector
$E$	: expectation
$F$	: vector defined in Eq. (77)
$F_k, G_k$	: solutions of Diophantine equation defined in Eq. (22)
$F_t$	: set of data up to and including time $t$
$H$	: matrix defined in Eq. (76)
$I_n$	: $n \times n$ identity matrix
$I_U, I_R$	: identity matrices defined in Eqs. (80) and (81)
$J, J_1$	: long-range cost-function
$M$	: polynomial matrix
$nA, nB, nC, nG_k, nV$	: order of model polynomial matrix $A, B, C, G_k, V$
$Nu$	: control horizon
$P$	: filter polynomial matrix
$PH$	: output prediction horizon
$Q$	: positive definite covariance matrix
$r$	: input weighting matrix
$R$	: input weighting matrix
$\hat{r}$	: filtered output signal vector
$\hat{s}f$	: optimal predictor vector of filtered output signal
$\hat{s}f$	: output prediction error vector
$SF, SF_r, \hat{S}F$	: optimal output predictor vector
$t$	: time
$u$	: input vector
$u_g$	: initial guess of $u$

$\tilde{U}$	: input vector
$w$	: setpoint vector
$w$	: setpoint
$W$	: setpoint vector
$y$	: output vector
$\hat{y}$	: output prediction
$Y$	: output matrix
$Y^*$	: past output vector
$\hat{Y}$	: differential output matrix
$Y\tilde{U}^d, \hat{Y}U^d$	: bilinear vector of optimal predictor vector
$Y^d$	: extended output matrix
$\hat{Y}^d$	: extended differential output matrix

## Greek Letters

$\alpha_s^d, \beta_k$	: predictor polynomial matrix
$\Delta$	: differencing operator
$\Delta U$	: input vector of optimal predictor vector
$\xi$	: interactor polynomial matrix
$\tau$	: minimum time delay
$v$	: normalized zero-mean white noise
$\Psi^d$	: model polynomial matrix

## REFERENCES

1. Mohler, R. R.: "Bilinear Control Processes", Academic Press, New York (1973).
2. Espana, M. and Landau, I. D.: *Automatica*, **14**, 345 (1978).
3. Svoronos, S., Stephanopoulos, G. and Aris, R.: *Int. J. Control*, **34**(4), 651(1981).
4. Cebuhar, W. A. and Costanza, V.: *Chemical Engineering Science*, **39**(12), 1715(1984).
5. Clarke, D. W., Mohtadi, C. and Tuffs, P. S.: *Automatica*, **23**(2), 137(1987).
6. Clarke, D. W., Mohtadi, C. and Tuffs, P. S.: *Automatica*, **23**(2), 149(1987).
7. Kinnaert, M.: *Int. J. Control*, **50**(1), 161(1989).
8. Favier, G. and Dubois, D.: *Automatica*, **26**(1), 75 (1990).
9. Clarke, D. W., Mohtadi, C. and Tuffs, P. S.: *Automatica*, **25**(6), 859(1989).
10. Press, W. H., Flannery, B. P., Teukolsky, S. A. and Vetterling, W. T.: "Numerical Recipes", Cambridge Univ. Press, Cambridge (1986).
11. Yeo, Y. K.: "Adaptive and Non-adaptive Bilinear Model Predictive Control", Ph.D. Dissertation, Auburn Univ., Auburn, Al, (1986).
12. Tao, G. and Ioannou, P. A.: *Int. J. Control*, **50**(4),

- 1153(1989).
13. 노균, 윤인섭, 여영구, 송형근 : 한국자동제어학술회의 논문집, **1**, 344(1990).
14. Dai, H. and Sinha, N. K.: *IEE Proc. D*, **136**(3), 122 (1989).
15. King, P. J., Burnham, K. J. and James, D. J. G.: *Control and Computers*, **18**(1), 4(1990).