

적응제어 시스템의 시변파라미터 추정에 관한연구
(Time-varying parameter estimation for adaptive control systems)

◦ 박상준, 전기준
(S. J. Park, G. J. Jeon)

경북대학교 공과대학 전자공학과
(Kyung Pook National Univ., Dept. of Elec. Eng.)

Abstract

This paper describes an efficient time-varying parameter estimation algorithm by resetting the parameter and P matrix of the RLS algorithm. The described algorithm is useful for estimating both jump parameter and drifting parameter which vary quite rapidly.

1 서 론

일반적으로 플랜트의 파라미터 값들을 모르거나 주변환경 또는 동작조건에 따라 플랜트의 동특성이 변화할 경우에도 스스로 적응하여 제어하는 방식을 적응제어라 한다. 이 적응제어에서 특히 플랜트의 파라미터가 시변일 경우에는 다음과 같은 사항들을 고려해야 한다.¹⁾ 2) 첫째, 시변파라미터(time-varying parameter)의 추정방법은 시불변파라미터의 추정방법과는 달리 파라미터 추정값이 어느한값으로 수렴되지 않고 연속적으로 시변파라미터의 값을 추정할수 있도록 하여야 한다. 둘째, 시변파라미터의 변화형태는 크게 jump 파라미터와 drifting 파라미터로 나눌수 있는데 각각의 경우에 대하여 적절한 추정방법을 선택하여야 한다. jump 파라미터 즉, 파라미터가 일정한 값을 가지다가 어느 순간 급격히 변화하게 되는 형태이다. 이러한 경우에는 축차최소자승(recursive least square) 알고리즘의 p 행렬을 주기적으로 초기화(resetting)시키는 방법³⁾ 이나 가변망각지수(variable forgetting factor)를 사용하는 방법⁴⁾ 등을 이용함으로써 해결 할수 있다. drifting 파라미터 즉, 비교적 느린 속도의 변화율로써 연속적으로 변화하는 형태이다. 이러한 경우에는 비선형적인 시변파라미터를 연속적으로 추정하여야 하므로 jump 파라미터의 경우와는 달리 망각지수(forgetting factor) λ 를 사용하는 방법²⁾이나 축차최소자승 알고리즘의 P행렬을 개선하는 방법⁵⁾ 등으로 해결할수있다. 그러나 이러한 기존의 시변파라미터 추정방법 들은 파라미터의 변화형태에 따라 적절한 추정방법 들을 선택하여야 하고 비교적 빠른변화를 가지는 drifting 파라미터의 경우에는 추정 불가능하며 jump 파라미터와 drifting 파라미터가 동시에 존재할 경우에도 시변파라미터 추정에 어려움이 있다는 단점들이 있다.

Xie 와 Evans⁶⁾ 는 시변파라미터들을 1차 근사화하여 모델링하고 주기적으로 시변파라미터와 P행렬

을 초기화 시켜줌으로써 기존의 알고리즘이 가지고있는 문제점들을 개선하였다. 그러나 이 알고리즘은 각 시스템에 적합한 초기화 주기(resetting period) T 를 선택하지 못하면 파라미터 추정성능 이 나빠지므로 미리 초기화 주기 T 를 알고 있어야한다는 문제점이 있다.

본 논문에서는 이러한 문제점을 해결하기 위하여 자동적으로 선택 된 가변 초기화주기(variable resetting period) T_v 를 사용함으로써 다양한 시스템에서도 적용 가능하고, 시변파라미터가 비교적 빠르게 변화하는 drifting 파라미터 경우 뿐만 아니라 파라미터 jump 현상이 존재하는 경우에도 추정가능한 시변파라미터 추정방법에 대해 연구한다.

2 시변 시스템의 모델링

시스템을 다음과같은 시변 ARMAX(auto regressive moving average with auxiliary input) 모델로 나타낼 수 있다고 가정한다.

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

여기서

$$\begin{aligned} A(z^{-1}) &= 1 + a_1(t)z^{-1} + \dots + a_n(t)z^{-n} \\ B(z^{-1}) &= b_1(t) + b_2(t)z^{-1} + \dots + b_m(t)z^{-m} \\ C(z^{-1}) &= 1 \end{aligned}$$

이며, z^{-1} 는 시간지연 연산자(backward shift operator) 이고 지연시간 k 는 알고 있다고 가정한다($k=1$). 그리고 $a_i(t), i=1, \dots, n, b_j(t), j=1, \dots, m$ 들은 미지의 시변파라미터들이고, $w(t)$ 는 백색가우스 잡음이며 각다항식의 차수 m, n 은 알고 있다고 가정한다.

ARMAX 모델을 풀어쓰면

$$\begin{aligned} y(t) &= -a_1(t)y(t-1) - \dots - a_n(t)y(t-n) \\ &+ b_1(t)u(t-1) + \dots + b_m(t)u(t-m) \\ &+ w(t) \end{aligned} \quad (2)$$

와 같이 나타나고 여기서 시변파라미터를 효과적으로 추정하기 위하여 이 파라미터들이 다음과 같이 변화한다고 가정하며 계산량을 줄이기 위하여 1차 근사화한다.

$$a_i(t) = a_{i0} + t_1 \cdot a_{i1} + t_1^2 \cdot a_{i2} + \dots \approx a_{i0} + t_1 \cdot a_{i1}$$

$$b_j(t) = b_{j0} + t_1 \cdot b_{j1} + t_1^2 \cdot b_{j2} + \dots \approx b_{j0} + t_1 \cdot b_{j1} \quad (3)$$

제어법칙을 유도하기 위하여 (3)식을 (2)식에 대입하면 다음과 같은 회귀식(regression) 형태로 나타낸다.

$$y(t) = - (a_{10} + t_1 \cdot a_{11})y(t-1) - \dots - (a_{n0} + t_1 \cdot a_{n1})y(t-n) + (b_{10} + t_1 \cdot b_{11})u(t-1) + \dots + (b_{m0} + t_1 \cdot b_{m1})u(t-1) \quad (4)$$

위 식을 데이터벡터와 파라미터 벡터로 표현하면

$$y(t) = \Phi(t-1)^T \theta(t-1) \quad (5)$$

와 같이 되고 파라미터 및 데이터 벡터는

$$\theta(t-1)^T = [\theta_1(t-1)^T; \theta_2(t-1)^T] \quad (6)$$

$$\theta_1(t-1)^T = [a_{10}, \dots, a_{n0}, b_{10}, \dots, b_{m0}]$$

$$\theta_2(t-1)^T = [a_{11}, \dots, a_{n1}, b_{11}, \dots, b_{m1}]$$

$$\Phi(t-1)^T = [\Phi_1(t-1)^T; t_1 \cdot \Phi_2(t-1)^T] \quad (7)$$

$$\Phi_1(t-1)^T = [y(t-1), \dots, y(t-n), u(t-1), \dots, u(t-m)]$$

$$\Phi_2(t-1)^T = [y(t-1), \dots, y(t-n), u(t-1), \dots, u(t-m)]$$

로 정의되고 $\theta_1(t-1)^T$ 및 $\theta_2(t-1)^T$ 는 $m+n$ 벡터이다. 이 방법은 $m+n$ 개의 파라미터를 추정하기 위해서는 (6)식에 나타난 것과 같이 $2(m+n)$ 개의 파라미터를 추정해야하므로 계산량이 두배로 증가하는 단점을 가지고 있다. 만일 어떤 파라미터들이 시불변이라는 것을 미리 알고 있을 경우에는 파라미터들을 1차 근사화하지 않고 상수로 둬므로써 계산량을 줄일 수 있다. 예를 들면 $a_i(t)$ 가 시불변파라미터 라는 것을 미리 알고 있으면 $a_i(t) = a_{i0} + t_1 \cdot a_{i1}$ 로 하는 대신 $a_i(t) = a_{i0}$ 로 둬므로써 추정할 파라미터 갯수를 줄일 수 있다.

3 하중 요소를 첨가한 최소분산형제어의 설계

ARMAX 모델 (1)의 one-step-ahead predictor 는 다음과 같다.

$$C(z^{-1})y^o(t+1|\mathcal{F}_t) = \alpha(z^{-1})y(t) + \beta(z^{-1})u(t) \quad (8)$$

여기서

$$y^o(t+1|t) = E\{y(t+1)|\mathcal{F}_t\} = y(t+1) - F(z^{-1})w(t+1) \quad (9)$$

$$\alpha(z^{-1}) = G(z^{-1})$$

$$\beta(z^{-1}) = F(z^{-1})B(z^{-1})$$

이고 $G(z^{-1})$ 와 $F(z^{-1})$ 는 다음의 Diophantine 방정식

$$C(z^{-1}) = F(z^{-1})A(z^{-1}) + z^{-k}G(z^{-1}) \quad (10)$$

를 만족하는 유일해가 되며

$$F(z^{-1}) = 1 + f_1 \cdot z^{-1} + \dots + f_{k-1} \cdot z^{-k+1}$$

$$G(z^{-1}) = g_0 + g_1 \cdot z^{-1} + \dots + g_{n-1} \cdot z^{-n+1}$$

이다.

ARMAX 모델 (1)의 비용함수(cost function)는

$$J = E\{1/2[y(t+1)-y^*(t+1)]^2 + Q/2 \cdot u(t)^2 | \mathcal{F}_t\} \quad (11)$$

로 정의한다.

(9)식과 (10)식을 이용하여 비용함수 J 를 최소화시키는 제어법칙을 구하면

$$u(t) = \frac{\beta_0}{\beta_0 + Q} \{z[\beta_0 - \beta(z^{-1})]u(t-1) + y^*(t+1) + [C(z^{-1}) - 1]y^o(t+1|t) - \alpha(z^{-1})y(t)\} \quad (12)$$

이다.

4 시변파라미터 추정방법

이미 1장에서 설명한 것과 같이 기존의 시변 파라미터 추정방법들은 파라미터의 변화형태에 따라 적절한 추정방법을 들을 선택하여야 하고 비교적 빠른 변화율을 가지는 drifting 파라미터의 경우에는 추정 불가능하며 jump 파라미터와 drifting 파라미터가 동시에 존재할 경우에도 파라미터 추정에 어려움이 있다는 단점들을 가지고 있다. 본 장에서는 이러한 문제점들을 해결할 수 있는 파라미터 추정방법에 대해서 설명한다.

먼저 파라미터 추정시 사용되는 변수들을 정의하면 t_1 은 시변파라미터들을 1차 근사화 하면서 생기는 변수로서 매 반복회수 마다 1 씩 증가시키고 가변 초기화주기 T_v 는 초기화되는 순간의 t_1 값을 대입한다. 먼저 망각지수가 포함된 축차최소자승 알고리즘을 이용하여 파라미터를 추정하면

$$e(t) = y(t) - \theta(t-1)^T \cdot \Phi(t-1) \quad (13)$$

$$\theta(t) = \theta(t-1) + \frac{P(t-2) \Phi(t-1) e(t)}{\lambda + \Phi(t-1)^T P(t-2) \Phi(t-1)}$$

$$P(t-1) = \frac{1}{\lambda} \left[P(t-2) \frac{P(t-2) \Phi(t-1) \Phi(t-1)^T P(t-2)}{\lambda + \Phi(t-1)^T P(t-2) \Phi(t-1)} \right]$$

이고 P 행렬의 초기치 $P(-1) = K_0 I$ 이고 망각지수 λ 는 0보다 큰 값이고 1 보다 작은 수이며, 여기서 $K_0 \geq 0$ 이고 I 는 $2(m+n) \times 2(m+n)$ 항등행렬이다. (13)식의 파라미터 추정방법을 계속 사용하다가 만일 $\Phi(t-1)^T P(t-1) \Phi(t-1) \leq \epsilon$ 이거나 $t_i \geq 10$ 일 경우에는 t_1 , 파라미터 및 P행렬들을 초기화 시킨다. 여기서 ϵ 은 아주 작은 양수값이다. 이렇게 하는 이유는 $\Phi(t-1)^T P(t-1) \Phi(t-1) \leq \epsilon$ 이면 추정값이 어느 한 값으로 수렴됨으로 연속적으로 변화하는 파라미터들을 추정하지 못하게 된다. 그러므로 이조건이 만족될 때 다시 초기화(resetting) 시켜줌으로서 연속적인 추정을 가능하게 한다. 그리고

t_1 이 계속적으로 증가하게 되면 데이터 벡터가 무한히 커져 계산이 불가능 해지므로 t_1 이 일정한 값 일반적으로 10 이상이 되면 초기화 시켜준다. t_1 및 파라미터 초기화방법은

$$\hat{\theta}(t) = \begin{bmatrix} \hat{\theta}_1(t) \\ \hat{\theta}_2(t) \end{bmatrix} = \begin{bmatrix} I_1 & T_v I_1 \\ 0 & I_1 \end{bmatrix} \begin{bmatrix} \hat{\theta}_1(t^-) \\ \hat{\theta}_2(t^-) \end{bmatrix} \quad (14)$$

이때 I_1 과 0는 $(m+n) \times (m+n)$ 항등행렬이고 t^- 는 초기화하기 바로전의 시간을 의미한다.

P행렬의 초기화방법은 확정적 시스템과 스토캐스틱 시스템에 대해서 각각 다른방법을 이용한다. 확정적 시스템의 경우에는

$$P(t-1) = I \quad (15)$$

이고 이때 I 는 $2(m+n) \times 2(m+n)$ 항등행렬이다. 스토캐스틱 시스템에서는 (15)식의 방법으로 초기화할 경우 잡음에 의한 영향으로 파라미터 추정성능이 매우 떨어지므로 다음과 같은 방법으로 P행렬을 초기화시킨다.

$$P(t-1) = \begin{bmatrix} I_1 & T_v \cdot I_1 \\ 0 & I_1 \end{bmatrix} P(t-1) \begin{bmatrix} I_1 & 0 \\ T_v \cdot I_1 & I_1 \end{bmatrix} \quad (16)$$

여기서 I_1 은 $(m+n) \times (m+n)$ 항등행렬이다.

5 시뮬레이션 결과 및 고찰

본 장에서는 제안한 시변파라미터 추정방법의 타당성을 확인하기 위하여 몇가지 예제에 대한 시뮬레이션과 고찰을 하였으며 이때 각 시뮬레이션들은 시변파라미터와 시불변파라미터 들이 동시에 존재하는 시스템 들이고 어떤 파라미터들이 시불변 인지지는 미리 알고 있다는 가정하에서 하기도한다.

5.1 확정적 시스템

(예제 1)

연속적으로 변화하는 시변파라미터 $a_1(t)$ 와 두개의 시불변파라미터 a_2, b_1 가 존재하는 시스템의 경우 시스템은

$$y(t) = a_1(t) * y(t-1) + a_2(t) * y(t-2) + b_1(t) * u(t-1)$$

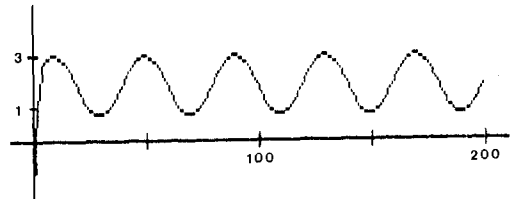
와같이 나타나며 이때 $a_1(t) = 2 + \sin(3.14 * t / 20)$, $a_2(t) = 2$, $b_1(t) = -3$ 이다. 시변파라미터는 1차 근사화 하고 시불변파라미터들은 상수로 모델링하여 다시 나타내면

$$y(t) = (a_{10} + t_1 a_{11}) * y(t-1) + a_2 * y(t-2) + b_1 * u(t-1)$$

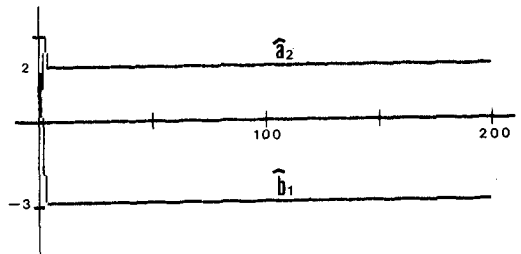
이 되고 파라미터 벡터와 데이터 벡터로 표현하면

$$\begin{aligned} \theta(t-1)^T &= [a_{10}, a_2, b_1, a_{11}] \\ \Phi(t-1)^T &= [y(t-1), y(t-2), u(t-1), t_1 * y(t-1)] \end{aligned}$$

이다. 그림1.에서는 시뮬레이션 결과를 나타내고있으며 그림1.(a)는 시변파라미터 $a_1(t)$ 의 추정결과이고 그림1.(b)는 두개의 시불변파라미터 a_2, b_1 의 추정결과를 나타낸다. 시뮬레이션 결과 비교적 빠른 변화율을 가지는 drifting 파라미터가 존재하는 경우에도 파라미터 추정이 잘됨을 알수있다.



(a) drifting 파라미터 $a_1(t)$ 의 추정치



(b) 시불변파라미터 a_2, b_1 의 추정치

그림 1. 예제1의 시뮬레이션 결과

(예제 2)

drifting 하는 시변파라미터 $a_1(t)$ 와 jump 하는 시변파라미터 $b_1(t)$ 가 동시에 존재하는 시스템에 대한 경우 시뮬레이션을 한다. 시스템을 나타내면

$$y(t) = a_1(t) * y(t-1) + a_2(t) * y(t-2) + b_1(t) * u(t-1)$$

$$\begin{aligned} \text{이때 } a_1(t) &= 2 + \sin(3.14 * t / 20), \quad a_2(t) = 2, \\ b_1(t) &= \begin{cases} -0.5 & 0 \leq t \leq 80 \\ 1.0 & 80 < t \leq 150 \\ 3.15 - 0.03t & 150 < t \leq 200 \end{cases} \end{aligned}$$

이다. 이 시스템에 대해서 시변파라미터들을 1차 근사화 하고 시불변파라미터는 상수로 모델링함으로써 다음과 같이 표현할 수 있다.

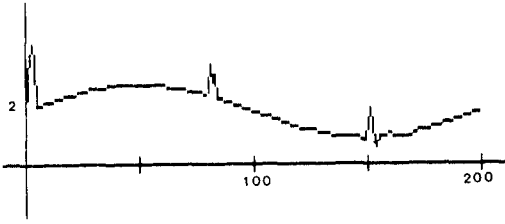
$$y(t) = (a_{10} + t_1 a_{11}) * y(t-1) + a_2 * y(t-2) + (b_{10} + t_1 b_{11}) * u(t-1)$$

위 식을 파라미터 벡터와 데이터 벡터로 나타내면

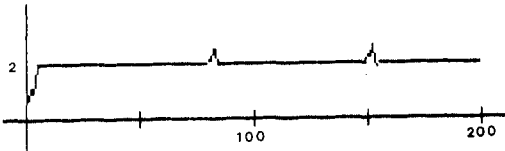
$$\begin{aligned} \theta(t-1)^T &= [a_{10}, a_2, b_{10}, a_{11}, b_{11}] \\ \Phi(t-1)^T &= [y(t-1), y(t-2), u(t-1), \\ &\quad t_1 * y(t-1), t_1 * u(t-1)] \end{aligned}$$

이다. 그림2.(a)는 drifting 파라미터에 대한 추정 결과를 나타내고 그림2.(b)는 시불변파라미터에 대한 추정 결과이고 그림2.(c)는 jump 파라미터에 대한 추정 결과를 나타내었으며 그림2.(d)는 jump 파라미터의

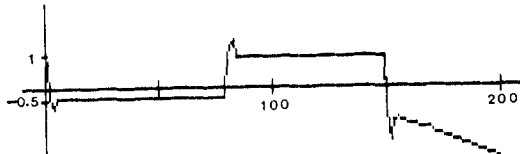
추정오차를 나타내었다. 시뮬레이션 결과 jump 파라미터의 영향으로 jump 하는 순간 모든 파라미터들에게 영향을 주어 약간의 진동이 발생된다. 그러나 빠른속도로 다시 추정됨을 알 수 있다.



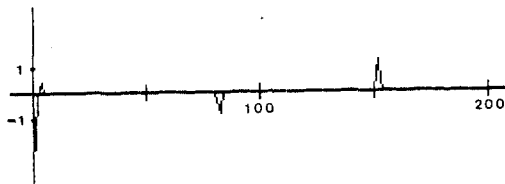
(a) drifting 파라미터 $a_1(t)$ 의 추정치



(b) 시불변파라미터 a_2 의 추정치



(c) jump 파라미터 $b_1(t)$ 의 추정치



(d) jump 파라미터 $b_1(t)$ 의 추정오차

그림2. 예제2의 시뮬레이션 결과

5.2 스토캐스틱 시스템의 경우

앞의 두예제를 통하여 제안한 시변파라미터 추정 방법이 확정적 시스템에서는 잘 동작함을 보였다. 본절에서는 백색가우스 잡음이 존재하는 스토캐스틱 시스템에 대한 시뮬레이션과 고찰을 통하여 제안한 시변파라미터 추정 방법의 타당성을 확인한다. 이때 사용한 백색가우스 잡음의 평균값이 영이고 공분산은 0.29 이다.

(예제 3)

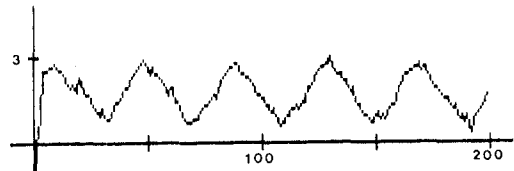
이미 앞에서 고찰한 (예제1)의 시스템에 측정 잡음 $w(t)$ 가 존재하는 경우 시스템은

$$y(t) = a_1(t)*y(t-1) + a_2(t)*y(t-2) + b_1(t)*u(t-1) + w(t)$$

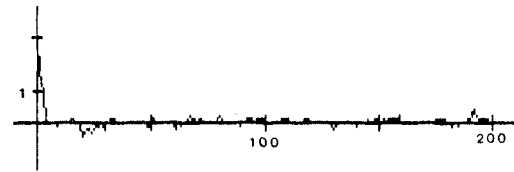
이며 이때 $a_1(t)=2+\sin(3.14*t/20)$, $a_2(t)=2$, $b_1(t)=-3$ 이다. 시변파라미터들은 1차 근사화 하며 시불변파라미터들은 상수로 두어 모델링하고 잡음들을 근사화 하여 다시 나타내면

$$y(t) = (a_{10} + t_1 * a_{11}) * y(t-1) + a_2 * y(t-2) + b_1 * u(t-1)$$

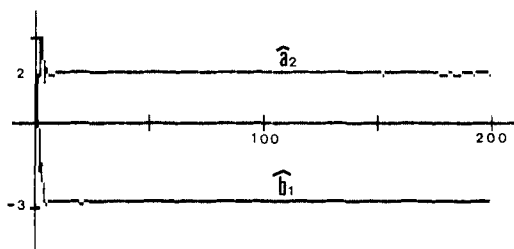
이다. 그림3.(a)는 drifting 파라미터 $a_1(t)$ 의 추정 결과를 나타내고 그림3.(b)는 $a_1(t)$ 의 추정오차를 나타내고 그림3.(c)는 시불변파라미터 a_2, b_1 의 추정 결과를 나타낸다. 시뮬레이션 결과 잡음에 의한 영향으로 약간의 추정오차가 발생하지만 잘 추정하고 있음을 알 수 있다.



(a) drifting 파라미터 $a_1(t)$ 의 추정치



(b) drifting 파라미터 $a_1(t)$ 의 추정오차



(c) 시불변파라미터 a_2, b_1 의 추정치

그림3. 예제3의 시뮬레이션 결과

(예제 4)

시불변파라미터 a_1, a_2 와 jump 하는 시변파라미터 $b_1(t)$ 로 이루어진 시스템에 측정잡음이 존재하는 경우에 대한 시뮬레이션. 시스템은

$$y(t) = a_1(t)*y(t-1) + a_2(t)*y(t-2) + b_1(t)*u(t-1) + w(t)$$

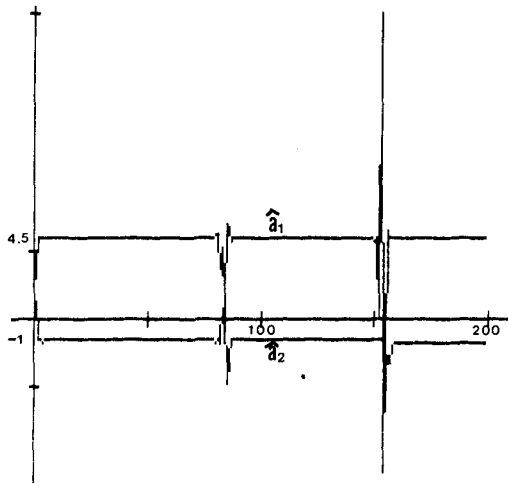
이며 $a_1(t) = 4.5$, $a_2(t) = -1$

$$b_1(t) = \begin{cases} -0.5 & 0 \leq t \leq 80 \\ 1.0 & 80 < t \leq 150 \\ -0.2 & 150 < t \leq 200 \end{cases}$$

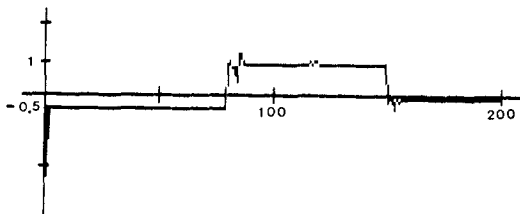
이다 이 시스템을 앞의 예제에서와 같은 방법으로 모델링하여 다시 나타내면

$$y(t) = a_1 * y(t-1) + a_2 * y(t-2) + (b_{10} + t_1 - b_{11}) * u(t-1)$$

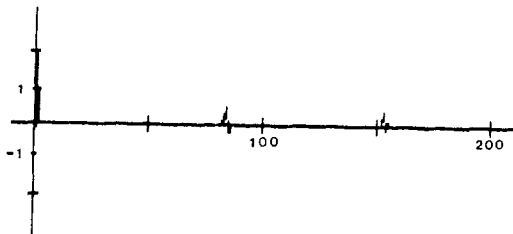
이다. 그림 4.(a)는 두개의 시불변파라미터 a_1, a_2 의 추정 결과이고 그림 4.(b)는 jump 파라미터의 추정결과를 나타내고 그림 4.(c)는 jump 파라미터의 추정오차를 나타낸다. 측정잡음과 jump 파라미터에 의한 영향으로 파라미터가 jump 하는 순간 모든 파라미터들이 추정을 하지 못하지만 곧 빠른속도로 다시 추정됨을 알수있다



(a) 시불변파라미터 a_1, a_2 의 추정치



(b) jump 파라미터 $b_1(t)$ 의 추정치



(c) jump 파라미터 $b_1(t)$ 의 추정오차

그림 4. 예제 4의 시뮬레이션 결과

본 논문에서는 시변파라미터들을 1차 근사화하여 모델링하고 가변초기화주기 T_0 를 자동적으로 선택하여 축차최소자승 알고리즘의 시변파라미터들과 P행렬을 초기화 시켜 줌으로써 기존의 방법에 비해 비교적 빠르게 변화하거나 jump 현상이 동시에 존재하는 경우에도 추정가능한 시변파라미터 추정 방법에 대해 연구하였다. 그러나 제안한 방법은 확정적 시스템과 스토캐스틱 시스템에 대해 P행렬 초기화방법이 서로 다르다. 그러므로 어느경우에나 일반적으로 적용할수 있는 시변파라미터 추정방법의 개발이 앞으로의 과제이다.

참 고 문 헌

- [1] Goodwin G.C. and Kwai Sang Sin, " Adaptive Filtering Prediction and Control ", Prentice Hall ,1983
- [2] K.J.Astrom and B.Wittenmark, " Adaptive Control ", Addison Wesley ,1989
- [3] Goodwin, Elliott and E.K.Teoh, " Deterministic convergence of a self-tuning regulator with covariance resetting ", IEE Proc, Vol.130, pp. 6-8, January, 1983
- [4] Cordero and mayne, " Deterministic convergence of a self-tuning regulator with variable forgetting factor ", IEE Proc., Vol.128, pp.19-23, January, 1981
- [5] E.G.Vogel and T.F.Edgar, " Application of an adaptive pole-zero placement controller to chemical processes with variable dead time ", Am. Control Conf., Washington D.C., 1982
- [6] Xie Xianya and Robin J. Evans, " Discrete-time Adaptive control for Deterministic Time-varying System ", Automatica, Vol. 20, No. 3, pp. 309-319, 1984
- [7] Xie Xianya and Robin J. Evans, " Discrete Time Stochastic Adaptive Control for Time-varying Systems ", IEEE Trans Aut. Control, Vol. AC-29, No .7, July, 1984