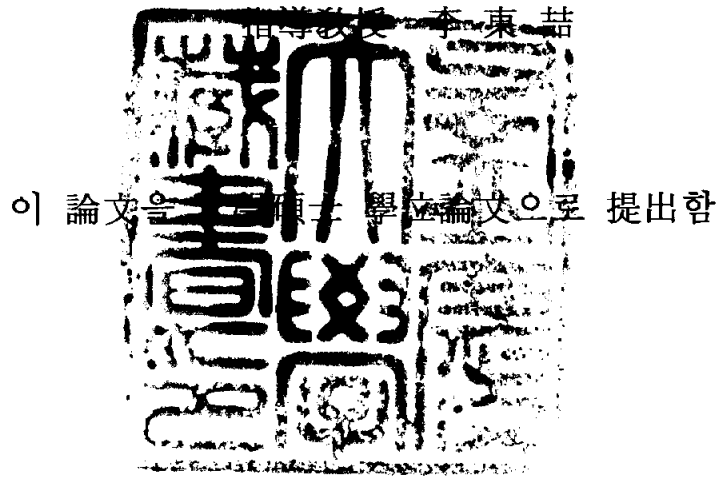


7/10
681
44
=5

工學碩士 學位論文

시스템동정의 ALS법에 관한 연구



2003年 2月

釜慶大學校 大學院

電氣工學科

李知烈

李知烈의 工學碩士 學位論文을 認准함

2002年 12月

主 審 工學 博士
委 員 工學 博士
委 員 工學 博士

洪 淳 一
裴 鍾 一
李 東 喆



목 차

제 1 장 서 론	1
제 2 장 최소2승법	3
2. 1 문제설정	3
2. 2 최소2승 추정치	5
2. 3 축차 최소2승법	6
2. 4 직교변환을 이용한 축차 최소2승법	9
제 3 장 TLS법	12
3. 1 문제설정	12
3. 2 TLS추정치	13
3. 3 RTLS알고리즘	14
제 4 장 ALS법	17
4. 1 문제설정	17
4. 2 ALS추정치	20
4. 3 RALS알고리즘	21
제 5 장 컴퓨터 시뮬레이션	23
5. 1 시스템 동정	23
5. 2 시뮬레이션 결과	25
제 6 장 결 론	39
참고문헌	40
Abstract	48

그림 목 차

Fig. 2. 1 Linear system model -----	3
Fig. 2. 2 System model with observed noise in output -----	4
Fig. 3. 1 System model with observed noise in input output -----	12
Fig. 4. 1 System model with observed noise in output -----	18
Fig. 5. 1 System model of simulation -----	24
Fig. 5. 2 Parameter estimation by RTLS method -----	25
Fig. 5. 3 Parameter estimation by RALS method -----	25
Fig. 5. 4 $y - \hat{y}_{TLS}$ -----	26
Fig. 5. 5 $y - \hat{y}_{ALS}$ -----	27
Fig. 5. 6 Parameter estimation by RTLS method -----	27
Fig. 5. 7 Parameter estimation by RALS method -----	28
Fig. 5. 8 $y - \hat{y}_{TLS}$ -----	29
Fig. 5. 9 $y - \hat{y}_{ALS}$ -----	29
Fig. 5. 10 Parameter estimation by RTLS method -----	30

Fig. 5. 11	Parameter estimation by RALS method -----	30
Fig. 5. 12	$y - \hat{y}_{TLS}$ -----	32
Fig. 5. 13	$y - \hat{y}_{ALS}$ -----	32
Fig. 5. 14	Parameter estimation by RTLS method -----	33
Fig. 5. 15	Parameter estimation by RALS method -----	33
Fig. 5. 16	$y - \hat{y}_{TLS}$ -----	35
Fig. 5. 17	$y - \hat{y}_{ALS}$ -----	35
Fig. 5. 18	Parameter estimation by RTLS method -----	36
Fig. 5. 19	Parameter estimation by RALS method -----	36
Fig. 5. 20	$y - \hat{y}_{ALS}$ -----	38

표 목 차

Table 1 Parameters and estimates	-----	26
Table 2 Parameters and estimates	-----	28
Table 3 Parameters and estimates	-----	31
Table 4 Parameters and estimates	-----	34
Table 5 Parameters and estimates	-----	37

제1장 서론

제어이론을 실제 시스템에 적용하기 위하여 먼저 제어대상인 시스템의 수학적 모델이 필요하다. 그러나 대부분의 경우 시스템의 수학적 모델은 미지이며 시간에 따라 변화한다. 따라서 시스템의 특성을 나타내는 수학적 모델을 구축하는 것을 동정(Identification)이라 하며 시스템 이론의 기본이 된다[1, 2, 3].

시스템동정(System identification)이란 적당한 입력을 대상 시스템에 주어 그 출력을 관측하고 그 입·출력 데이터에 기초하여 시스템의 수학적 모델을 구축하는 것이다[4, 5].

시스템의 동정문제는 다음의 3단계로 분류할 수 있다.

- 1) 모델구조의 가정
- 2) 입·출력 데이터로부터 미지 파라미터의 추정
- 3) 모델의 타당성 검증

종래의 시스템제어 분야에 있어서 동정문제의 대부분은 모델구조(차분방정식, 전달함수의 형)를 미리 정해 두고 모델에 포함되는 미지 파라미터를 추정하는 문제로 취급했다. 파라미터 추정[6]에 대하여 지금까지 여러 가지 방법이 제안되고 있지만, 그 대표적인 방법이 최소 2승법(Least squares method, LS)이다[7].

시스템 동정에 있어서 입·출력에 관측 잡음이 없는 경우는 LS법에 의해 파라미터 참값을 구할 수 있다.

한편, 입·출력에 관측잡음이 있는 경우는 TLS(Total least squares, TLS)법[8, 9, 10]에 의해 파라미터의 일치추정치(Consistent estimate)를 LS법과 TLS법의 중간적인 방법으로서 ALS(Adjusted least squares, ALS)법을 제안한다.

본 논문에서는 일반적으로 저주파 대역에서 부적합하고 어느 정도 잡음이 크면 바이어스(Bias)를 무시할 수 없는 종래의 최소2승법을 개선하여 ALS법에 대해 검토하고 컴퓨터 시뮬레이션에 의해 추정결과를 논하였다.

본 논문의 구성은 다음과 같다. 1장은 서론이며 2장은 최소2승법, 축차최소2승법(Recursive least squares method, RLS)을 이용하여 축차계산 알고리즘을 도출하고, 3장은 TLS법을 이용하여 축차계산 알고리즘을 도출한다. 4장은 입력에는 관측잡음이 없고 출력에만 관측잡음이 있는 경우의 파라미터의 일치추정치가 얻어지는 ALS법에 대해 논하였고, 5장은 시스템모델을 설정하여 RTLS(Recursive total least squares)법과 RALS(Recursive adjusted least squares)추정법을 이용한 파라미터 추정과 시뮬레이션 결과를 알아보고, 6장은 결론이다.

제2장 최소2승법

본 장에서는 최소2승법, 축차 최소2승법을 이용하여 축차계산 알고리즘을 도출한다.

2. 1 문제 설정

Fig. 2.1에 나타낸 단일입력 단일출력의 선형 시스템을 고려하자.

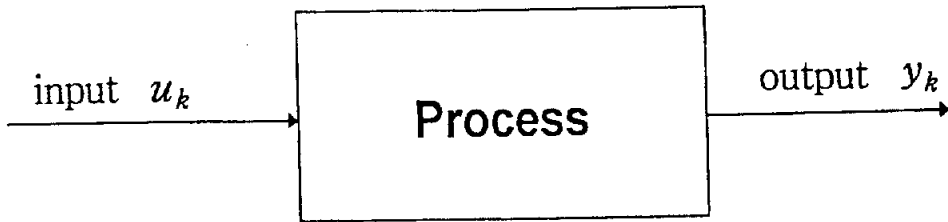


Fig. 2. 1 Linear system model

이 시스템은 다음 차분방정식으로 나타낼 수 있다.

$$y_k = \beta_0 u_k + \beta_1 u_{k-1} + \cdots + \beta_n u_{k-n} - \alpha_1 y_{k-1} - \cdots - \alpha_n y_{k-n} \quad (2.1)$$

여기서, 벡터를 식(2.2)와 식(2.3)과 같이 두면 식(2.1)은 식(2.4)와 같이 나타낼 수 있다.

$$\mathbf{m}_{Ak}^T = [u_k, u_{k-1}, \cdots, u_{k-n}, -y_{k-1}, \cdots, -y_{k-n}] \quad (2.2)$$

$$\mathbf{x}^T = [\beta_0, \beta_1, \dots, \beta_n, \alpha_1, \dots, \alpha_n] \quad (2.3)$$

$$y_k = \mathbf{m}_{Ak}^T \mathbf{x} \quad (2.4)$$

또 행렬벡터를 식(2.5), (2.6)과 같이 두면 식(2.1)의 관계는 식(2.7)과 같이 나타낼 수 있다.

$$\mathbf{A}_m^T = [\mathbf{m}_{Ak+1}, \mathbf{m}_{Ak+2}, \dots, \mathbf{m}_{Ak+m}] \quad (2.5)$$

$$\mathbf{b}^T = [y_{k+1}, y_{k+2}, \dots, y_{k+m}] \quad (2.6)$$

$$[\mathbf{A}_m \quad -\mathbf{b}_m] \begin{bmatrix} \mathbf{x} \\ 1 \end{bmatrix} = 0 \quad (2.7)$$

출력에 관측잡음이 있는 경우의 입·출력관계를 Fig. 2.2에 나타내었다.

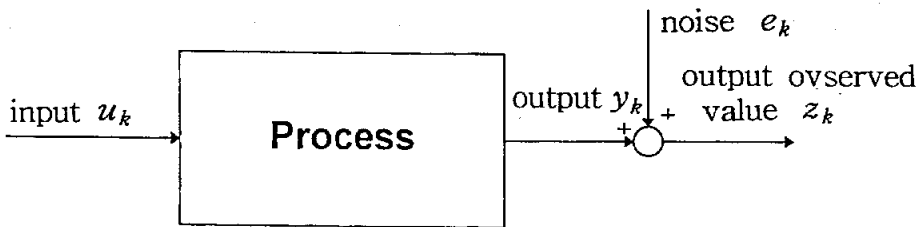


Fig. 2. 2 System model with observed noise in the output

관측잡음이 있는 경우의 벡터 \mathbf{b}_m 을 식(2.8)과 같이 정의하면 식(2.9)를 얻을 수 있다.

$$\mathbf{b}_m^T = [z_{k+1}, z_{k+2}, \dots, z_{k+m}] \quad (2.8)$$

$$[\mathbf{A}_m \quad -\mathbf{b}_m] \begin{bmatrix} \mathbf{x} \\ 1 \end{bmatrix} \cong 0 \quad (2.9)$$

2. 2 최소 2승추정치

입력열 u_k , 출력열 z_k 가 관측되었다고 하자. 최소2승법은 식(2.9)에 대해 식(2.10)을 최소로 하는 \mathbf{x} 를 구한다. 식(2.10)의 양변을 \mathbf{x} 로 미분하면 식(2.11)이 된다.

$$J = [\mathbf{b} - A\mathbf{x}]^T [\mathbf{b} - A\mathbf{x}] = \|\mathbf{b} - A\mathbf{x}\|_2^2 \quad (2.10)$$

$$\begin{aligned} \frac{\partial J}{\partial \mathbf{x}} &= \frac{\partial([\mathbf{b} - A\mathbf{x}]^T [\mathbf{b} - A\mathbf{x}])}{\partial \mathbf{x}} \\ &= 2 \frac{\partial([\mathbf{b} - A\mathbf{x}]^T)}{\partial \mathbf{x}} [\mathbf{b} - A\mathbf{x}] \\ &= -2A^T [\mathbf{b} - A\mathbf{x}] \end{aligned} \quad (2.11)$$

$$\left. \frac{\partial J}{\partial \mathbf{x}} \right|_{\mathbf{x} = \hat{\mathbf{x}}_{LS}} = 0 \quad (2.12)$$

$$-2A^T [\mathbf{b} - A\hat{\mathbf{x}}_{LS}] = 0 \quad (2.13)$$

따라서, 미지 파라미터의 최소 2승추정치는 식(2.14)가 된다.

$$\hat{\mathbf{x}}_{LS} = (A^T A)^{-1} A^T \mathbf{b} \quad (2.14)$$

또, $\hat{\mathbf{x}}_{LS}$ 를 다른 방법으로 도출해 보자.

$$\begin{aligned} J &= [\mathbf{b} - A\mathbf{x}]^T [\mathbf{b} - A\mathbf{x}] \\ &= [\mathbf{b} - \mathbf{x}^T A^T] [\mathbf{b} - A\mathbf{x}] \\ &= \mathbf{x}^T A^T A \mathbf{x} - \mathbf{x}^T A^T \mathbf{b} - \mathbf{b}^T A \mathbf{x} + \mathbf{b}^T \mathbf{b} \end{aligned}$$

$$\begin{aligned}
&= \mathbf{x}^T A^T A \mathbf{x} - \mathbf{x}^T A^T A (A^T A)^{-1} A^T \mathbf{b} \\
&\quad - \mathbf{b}^T A^T A (A^T A)^{-1} A^T \mathbf{x} + \mathbf{b}^T A (A^T A)^{-1} A^T \mathbf{b} \\
&\quad - \mathbf{b}^T A (A^T A)^{-1} A^T \mathbf{b} + \mathbf{b}^T \mathbf{b} \\
&= (\mathbf{x}^T - \mathbf{b}^T A (A^T A)^{-1} A^T) A^T A (\mathbf{x} - (A^T A)^{-1} A^T \mathbf{b}) \\
&\quad + \mathbf{b}^T \mathbf{b} - \mathbf{b}^T A (A^T A)^{-1} A^T \mathbf{b} \\
&= (\mathbf{x} - (A^T A)^{-1} A^T \mathbf{b})^T A^T A (\mathbf{x} - (A^T A)^{-1} A^T \mathbf{b}) \\
&\quad + \mathbf{b}^T \mathbf{b} - \mathbf{b}^T A (A^T A)^{-1} A^T \mathbf{b} \tag{2.15}
\end{aligned}$$

따라서, $A^T A > 0$ 이면 \mathbf{x} 가 식(2.16)과 같이 표현될 때 J 는 최소가 되어 그 최소치는 식(2.17)과 같다.

$$\mathbf{x} = (A^T A)^{-1} A^T \mathbf{b} \tag{2.16}$$

$$J_{\min} = \mathbf{b}^T \mathbf{b} - \mathbf{b}^T A (A^T A)^{-1} A^T \mathbf{b} \tag{2.17}$$

즉, 식(2.14)와 같은 결과가 얻어진다.

2.3 축차 최소2승법[11]

앞 절에서 최소2승추정치에 대해 유도했다. 이 방법은 입력신호를 관측하여 기록해두고 미지 파라미터를 계산하는 동정법이며 오프-라인(Off-line)법이라 한다. 그러나 시시각각 데이터가 추가되는 경우 또는 시스템 파라미터가 시간과 함께 느리게 변하는 경우에는 오프-라인법으로 파라미터에 의한 추정치의 계산은 곤란하다.

여기서 현재 추정치를 그 직전의 추정치와 새로운 데이터로부터 축차적으로 갱신하면서 계산하는 온-라인(On-line)법을 고려하자.

한번 더 식(2.9)를 나타내자.

$$[A_m \quad -\mathbf{b}_m] \begin{bmatrix} \mathbf{x} \\ 1 \end{bmatrix} \cong 0 \quad (2.18)$$

k 시각에서 새로운 데이터가 추가된다면 $k-1$ 시각과 k 시각의 행렬 $[A_m \quad -\mathbf{b}_m]$ 사이의 관계는 식(2.19)와 같다.

$$[A_k \quad -\mathbf{b}_k] = \begin{bmatrix} A_{k-1} & -\mathbf{b}_{k-1} \\ \mathbf{m}_{Ak}^T & -z_k \end{bmatrix} \quad (2.19)$$

식(2.14)에서

$$\hat{\mathbf{x}}_{LS(k)} = (A_k^T A_k)^{-1} A_k^T \mathbf{b}_k \quad (2.20)$$

여기서, P_k 를 식(2.21)과 같이 정의하면 식(2.22)가 된다.

$$P_k = (A_k^T A_k)^{-1} \quad (2.21)$$

$$\begin{aligned} &= ([A_{k-1}^T \quad \mathbf{m}_{Ak}] \begin{bmatrix} A_{k-1} \\ \mathbf{m}_{Ak}^T \end{bmatrix})^{-1} \\ &= (A_{k-1}^T A_{k-1} + \mathbf{m}_{Ak} \mathbf{m}_{Ak}^T)^{-1} \\ &= (P_{k-1}^{-1} + \mathbf{m}_{Ak} \mathbf{m}_{Ak}^{-T})^{-1} \end{aligned} \quad (2.22)$$

식(2.22)에는 역행렬의 계산이 포함되어 있으므로 역행렬보제 (Matrix inversion lemma) 식(2.23)을 이용하여 식(2.22)를 변형하면 식(2.24)와 같이 나타낼 수 있다.

$$(A + BC)^{-1} = A^{-1} - A^{-1}B(I + CA^{-1}B)^{-1}CA^{-1} \quad (2.23)$$

$$P_k = P_{k-1} - \frac{P_{k-1} \mathbf{m}_{Ak} \mathbf{m}_{Ak}^T P_{k-1}}{1 + \mathbf{m}_{Ak}^T P_{k-1} \mathbf{m}_{Ak}} \quad (2.24)$$

식(2.20)에 식(2.19), (2.21), (2.24)를 대입하면 식(2.25)와 같이 표현된다.

$$\begin{aligned}
\hat{\mathbf{x}}_{LS(k)} &= (A_k^T A_k)^{-1} A_k^T \mathbf{b}_k \\
&= P_k A_k^T \mathbf{b}_k \\
&= P_{k-1} \left(1 - \frac{\mathbf{m}_{Ak} \mathbf{m}_{Ak}^T P_{k-1}}{1 + \mathbf{m}_{Ak}^T P_{k-1} \mathbf{m}_{Ak}} \right) \begin{bmatrix} A_{k-1} \\ \mathbf{m}_{Ak}^T \end{bmatrix}^T \begin{bmatrix} \mathbf{b}_{k-1} \\ z_k \end{bmatrix} \\
&= P_{k-1} \left(1 - \frac{\mathbf{m}_{Ak} \mathbf{m}_{Ak}^T P_{k-1}}{1 + \mathbf{m}_{Ak}^T P_{k-1} \mathbf{m}_{Ak}} \right) (A_{k-1} \mathbf{b}_{k-1} + z_k \mathbf{m}_{Ak}^T) \\
&= P_{k-1} A_{k-1} \mathbf{b}_{k-1} \\
&\quad + \frac{z_k P_{k-1} \mathbf{m}_{Ak} - P_{k-1} \mathbf{m}_{Ak} \mathbf{m}_{Ak}^T P_{k-1} A_{k-1} \mathbf{b}_{k-1}}{1 + \mathbf{m}_{Ak}^T P_{k-1} \mathbf{m}_{Ak}} \\
&= \hat{\mathbf{x}}_{LS(k-1)} \\
&\quad + \frac{P_{k-1} \mathbf{m}_{Ak}}{1 + \mathbf{m}_{Ak}^T P_{k-1} \mathbf{m}_{Ak}} (z_k - \mathbf{m}_{Ak}^T \hat{\mathbf{x}}_{LS(k-1)}) \quad (2.25)
\end{aligned}$$

식(2.24)과 식(2.25)는 축차 최소2승법의 알고리즘이다. 망각계수 (Forgetting factor) φ 를 이용하면 축차 최소2승법의 알고리즘은 식 (2.26), (2.27)과 같다.

$$\begin{aligned}
\hat{\mathbf{x}}_{LS(k)} &= \hat{\mathbf{x}}_{LS(k-1)} \\
&\quad + \frac{P_{k-1} \mathbf{m}_{Ak}}{\varphi + \mathbf{m}_{Ak}^T P_{k-1} \mathbf{m}_{Ak}} (z_k - \mathbf{m}_{Ak}^T \hat{\mathbf{x}}_{LS(k-1)}) \quad (2.26)
\end{aligned}$$

$$P_k = \frac{1}{\varphi} \left(I - \frac{P_{k-1} \mathbf{m}_{Ak} \mathbf{m}_{Ak}^T}{1 + \mathbf{m}_{Ak}^T P_{k-1} \mathbf{m}_{Ak}} \right) P_{k-1} \quad (2.27)$$

φ 값은 $0 \leq \varphi \leq 1$ 이지만, 일반적으로 1에 가까운 값, 예를 들면 0.95 또는 0.98이 이용된다.

2. 4 직교변환(Orthogonal transformation)을 이용한 축차 최소2승법

여기서 행렬 $[A \quad -b]$ 의 QR분해[12]를 고려하자.

$$Q^T[A \quad -b] = \begin{bmatrix} R_A & \gamma \\ 0 & \gamma \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} R \\ 0 \end{bmatrix} \quad (2.28)$$

단, R_A , R 는 상삼각행렬(Upper triangular matrix)이며, Q 는 직교행렬(Orthogonal matrix)이다[13, 14].

$k-1$ 시각과 k 시각의 행렬 $[A_m \quad -b_m]$ 사이의 관계는 식(2.29)와 같다.

$$[A_{k-1} \quad -b_k] = \begin{bmatrix} A_{k-1} & -b_{k-1} \\ \mathbf{m}_{Ak}^T & -z_k \end{bmatrix} \quad (2.29)$$

그리고 벡터 \mathbf{m}_k 를 식(2.30)과 같이 두고 행렬 $\begin{bmatrix} R_{k-1} \\ \mathbf{m}_k^T \end{bmatrix}$ 의 직교변환을 행하면 식(2.31)과 같이 나타낼 수 있다.

$$\mathbf{m}_k = \begin{bmatrix} \mathbf{m}_{Ak} \\ -z_k \end{bmatrix} \quad (2.30)$$

$$H \begin{bmatrix} R_{k-1} \\ \mathbf{m}_k^T \end{bmatrix} = \begin{bmatrix} R_k \\ 0 \end{bmatrix} = \begin{bmatrix} R_{Ak} & \gamma_k \\ 0 & \gamma \\ 0 & 0 \end{bmatrix} \quad (2.31)$$

여기서, H 는 직교행렬이다.

식(2.31)은 R_k 의 축차계산식이지만, 여기서 R_k^{-1} 의 축차계산을 고려하자.

벡터 \mathbf{S} 를 식(2.32)와 같이 정의하고, 식(2.33)을 만족하는 직교행렬 G 를 구한다.

$$\mathbf{S} = -R_{k-1}^{-T} \mathbf{m}_k \quad (2.32)$$

$$G \begin{bmatrix} \mathbf{S} \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ \delta \end{bmatrix}; GG^T = \mathbf{I}, \quad \delta = (1 + \|\mathbf{S}\|_2^2)^{\frac{1}{2}} \quad (2.33)$$

R_k^{-1} 는 식(2.34)에 의해 얻어진다.

$$G \begin{bmatrix} R_{k-1}^{-T} \\ 0 \end{bmatrix} = \begin{bmatrix} R_k^{-T} \\ \mathbf{u}^T \end{bmatrix}, \quad \mathbf{u} = R_{k-1}^{-1} \mathbf{S} / \delta \quad (2.34)$$

알고리즘 2. 1

행렬 R_{k-1}^{-1} 과 벡터 \mathbf{m}_k 를 기지라고 하자.

1. 식(2.32), (2.33), (2.34)에 의해 R_k^{-1} 을 계산한다.
2. 다음 식(2.35)에서 파라미터 벡터를 구한다.

$$\begin{bmatrix} \hat{\mathbf{x}}_{LS(k)} \\ 1 \end{bmatrix} = \frac{1}{\gamma_k^*} \begin{bmatrix} \gamma_k^* \\ \gamma_k^* \end{bmatrix} \quad (2.35)$$

$$\text{단, } R_k^{-1} = \begin{bmatrix} R_{Ak}^* & \gamma_k^* \\ 0 & \gamma_k^* \end{bmatrix} = \begin{bmatrix} R_{Ak}^{-1} & -R_{Ak}^{-1} \gamma_k \gamma_k^{-1} \\ 0 & \gamma_k^{-1} \end{bmatrix} \quad (2.36)$$

알고리즘 2. 2

행렬 R_{k-1}^{-1} 과 벡터 m_k 를 기지라고 하자.

1. 식(2.31)에 의해 R_k 를 계산한다.
2. 식(2.35), (2.36)에 의해 파라미터 벡터를 구한다.

제3장 TLS법

본 장에서는 입·출력 모두 관측잡음이 있는 경우에 파라미터의 일치추정치를 얻을 수 있는 TLS법을 이용하여 축차계산 알고리즘(RTLS)을 도출한다.

3. 1 문제설정

입·출력에 관측잡음이 있는 경우의 시스템을 Fig. 3.1에 나타내었다.

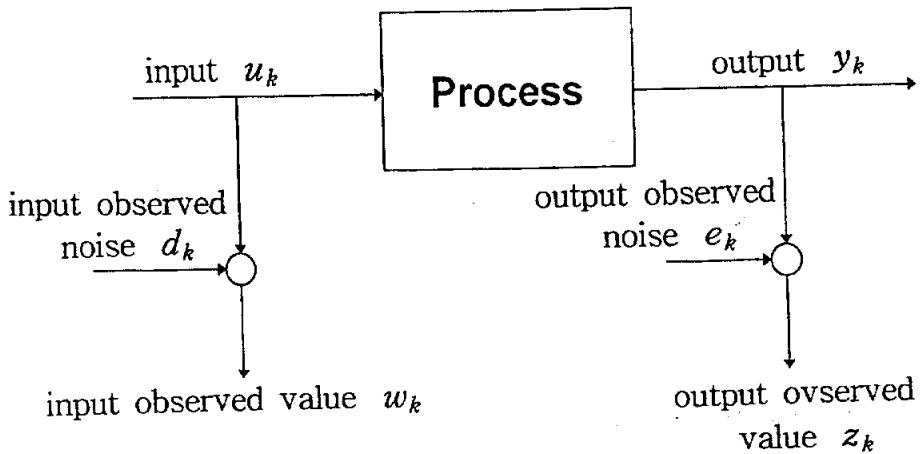


Fig. 3. 1 System model with observed noise in the input-output

벡터와 행렬을 식(3.1)~(3.4)와 같이 정의한다.

$$\mathbf{x}^T = [\beta_0, \beta_1, \dots, \beta_n, \alpha_1, \dots, \alpha_n] \quad (3.1)$$

$$\mathbf{m}_{Ak}^T = [w_k, w_{k-1}, \dots, w_{k-n}, -z_{k-1}, \dots, -z_{k-n}] \quad (3.2)$$

$$A_k^T = [\mathbf{m}_{An}, \dots, \mathbf{m}_{Ak-1}, \mathbf{m}_{Ak}] \quad (3.3)$$

$$\mathbf{b}_k^T = [z_n, z_{n+1}, \dots, z_k] \quad (3.4)$$

TLS법은 식(3.5)와 같다.

$$[A \quad -\mathbf{b}] \begin{bmatrix} \mathbf{x} \\ 1 \end{bmatrix} \cong 0 \quad (3.5)$$

$$\text{단, } A \in \mathbb{R}^{m \times p}, \mathbf{b} \in \mathbb{R}^m, \mathbf{x} \in \mathbb{R}^p, p = 2n + 1$$

어떤 행렬 $E \in \mathbb{R}^{m \times p}$ 와 벡터 $\boldsymbol{\gamma} \in \mathbb{R}^m$ 에 대해 식(3.6)을 만족하고 $\| [E, \boldsymbol{\gamma}] \|_F$ 를 최소로 하는 E 와 $\boldsymbol{\gamma}$ 를 각각 E_0 와 $\boldsymbol{\gamma}_0$ 라 하면 식 (3.7)을 만족하는 $\hat{\mathbf{x}}_{TLS}$ 를 구하는 방법이다.

$$\mathbf{b} + \boldsymbol{\gamma} \in \text{range}(A + E) \quad (3.6)$$

$$(A + E_0) \hat{\mathbf{x}}_{TLS} = \mathbf{b} + \boldsymbol{\gamma}_0 \quad (3.7)$$

3. 2 TLS추정치

행렬 $[A \quad -\mathbf{b}]$ 가 식(3.8)과 같은 특이치분해(Singular value decomposition, SVD)[15]가 된다.

$$U^T [A \quad -\mathbf{b}] V = \begin{bmatrix} \Sigma \\ 0 \end{bmatrix} \quad (3.8)$$

$$\text{단, } U^T U = I, V^T V = I, \Sigma = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_{p+1})$$

$$\sigma_1 \geq \dots \geq \sigma_p \geq \sigma_{p+1}$$

이때 행렬 V 를 식(3.9), (3.10)과 같이 분할하면 파라미터 추정치는 식(3.11)에서 구할 수 있다.

$$V = [\mathbf{v}_1, \dots, \mathbf{v}_p, \mathbf{v}_{p+1}] = [V_1, \mathbf{v}_{p+1}] \quad (3.9)$$

$$\mathbf{v}_{p+1} = [v_1, \dots, v_{p+1}] \quad (3.10)$$

$$\begin{bmatrix} \hat{\mathbf{x}}_{TLS} \\ 1 \end{bmatrix} = \frac{1}{v_{p+1}} \mathbf{v}_{p+1} \quad (3.11)$$

3.3 RTLS알고리즘

여기서, 행렬 $[A \quad -\mathbf{b}]$ 의 QR분해를 고려하자.

$$Q^T [A \quad -\mathbf{b}] = \begin{bmatrix} R_A & \boldsymbol{\gamma} \\ 0 & \gamma \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} R \\ 0 \end{bmatrix} \quad (3.12)$$

단, R_A , R 는 상삼각행렬이며, Q 는 직교행렬이다.

식(2.19)에 의해 $k-1$ 시각과 k 시각의 행렬 $[A_m \quad -\mathbf{b}_m]$ 사이의 관계는 식(3.13)과 같다.

$$[A_k \quad -\mathbf{b}_k] = \begin{bmatrix} A_{k-1} & -\mathbf{b}_{k-1} \\ \mathbf{m}_{Ak}^T & -z_k \end{bmatrix} \quad (3.13)$$

그리고 벡터 \mathbf{m}_k 를 식(3.14)와 같이 두고, 행렬 $\begin{bmatrix} R_{k-1} \\ \mathbf{m}_k^T \end{bmatrix}$ 의 직교

변환을 행하면 식(3.15)와 같다.

$$\mathbf{m}_k = \begin{bmatrix} \mathbf{m}_{Ak} \\ -z_k \end{bmatrix} \quad (3.14)$$

$$H \begin{bmatrix} R_{k-1} \\ \mathbf{m}_k^T \end{bmatrix} = \begin{bmatrix} R_k \\ 0 \end{bmatrix} = \begin{bmatrix} R_{Ak} & \boldsymbol{\gamma}_k \\ 0 & \gamma_k \\ 0 & 0 \end{bmatrix} \quad (3.15)$$

단, H 는 직교행렬이다.

한편, R_k^{-1} 의 축차계산은 식(3.16), (3.17), (3.18)과 같이 행해진다.

벡터 \mathbf{S} 는 식(3.16)과 같이 정의하고, 식(3.17)을 만족하는 직교행렬 G 를 구하고 R_k^{-1} 은 식(3.18)에 의해 구할 수 있다.

$$\mathbf{S} = -R_{k-1}^{-1} \mathbf{m}_k \quad (3.16)$$

$$G \begin{bmatrix} \mathbf{S} \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ \delta \end{bmatrix}, \quad G^T G = I, \quad \delta = (1 + \|\mathbf{S}\|_2^2)^{\frac{1}{2}} \quad (3.17)$$

$$G \begin{bmatrix} R_{k-1}^{-T} \\ 0 \end{bmatrix} = \begin{bmatrix} R_k^{-T} \\ \mathbf{u}^T \end{bmatrix}, \quad \mathbf{u} = R_{k-1}^{-1} \mathbf{S} / \delta \quad (3.18)$$

RTLS알고리즘은 다음의 2가지가 있다.

알고리즘 3. 1

행렬 R_{k-1}^{-1} 과 벡터 \mathbf{m}_k , $\hat{\mathbf{x}}_{TLS(k-1)}$ 은 기지라고 한다.

1. 식(3.16), (3.17), (3.18)에 의해 R_k^{-1} 을 계산한다.
2. 식(3.19), (3.20)에서 파라미터 벡터를 구한다.

초기치는 $\hat{\mathbf{x}}_{TLS(k)}^{(0)} = \hat{\mathbf{x}}_{TLS(k-1)}$ 라 둔다.

$$\mathbf{W} = \begin{bmatrix} \mathbf{w}_1 \\ w \end{bmatrix} = R_k^{-1} R_k^{-T} \begin{bmatrix} \hat{\mathbf{x}}_k^{(i)} \\ 1 \end{bmatrix} \quad (3.19)$$

$$\begin{bmatrix} \hat{\mathbf{x}}_k^{(i+1)} \\ 1 \end{bmatrix} = \frac{1}{w} \mathbf{w} \quad (3.20)$$

알고리즘 3. 2

행렬 R_{k-1} 과 벡터 \mathbf{m}_k 는 기지라고 한다.

1. 식(3.15)에 의해 R_k 를 계산한다.
2. 식(3.21)에 의해 R_k 의 SVD를 계산한다.

$$U^T R_k V = \begin{bmatrix} \Sigma \\ 0 \end{bmatrix} \quad (3.21)$$

단, $U^T U = I$, $V^T V = I$, $\Sigma = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_{p+1})$

$$\sigma_1 \geq \dots \geq \sigma_p \geq \sigma_{p+1}$$

3. 행렬 V 를 식(3.22), (3.23)과 같이 분할하고, 파라미터 TLS추정치를 식(3.24)에서 구한다.

$$V = [\mathbf{v}_1, \dots, \mathbf{v}_p, \mathbf{v}_{p+1}] = [V_1, \mathbf{v}_{p+1}] \quad (3.22)$$

$$\mathbf{v}_{p+1} = [\mathbf{v}_1, \dots, \mathbf{v}_{p+1}] \quad (3.23)$$

$$\begin{bmatrix} \hat{\mathbf{x}}_{TLS(k)} \\ 1 \end{bmatrix} = \frac{1}{v_{p+1}} \mathbf{v}_{p+1} \quad (3.24)$$

제4장 ALS법

본 장에서는 입력에는 관측잡음이 없고 출력에만 관측잡음이 있는 경우에 있어서 파라미터의 일치추정치가 얻어지는 ALS법을 취급하여 그 축차계산알고리즘을 도출한다.

4. 1 문제설정

$$[A \quad -b] \begin{bmatrix} \mathbf{x} \\ 1 \end{bmatrix} \cong 0 \quad (4.1)$$

단, $A \in \mathbf{R}^{m \times p}$, $b \in \mathbf{R}^m$, $\mathbf{x} \in \mathbf{R}^p$, $p = 2n + 1$

여기서, 식(4.1)의 최적해를 구하는 방법을 고려하고, 다시 한번 LS법과 TLS법의 정의를 고려한다.

LS법(관측잡음이 없는 경우)

어떤 벡터 $\boldsymbol{\gamma} \in \mathbf{R}^m$ 에 대하여 식(4.2)를 만족하고 $\|\boldsymbol{\gamma}\|_2$ 를 최소화하는 $\boldsymbol{\gamma}$ 를 $\boldsymbol{\gamma}_0$ 라 하면 식(4.3)을 만족하는 $\hat{\mathbf{x}}_{LS}$ 를 구한다.

$$\mathbf{b} + \boldsymbol{\gamma} \in \text{range}(A) \quad (4.2)$$

$$[A \quad -(\mathbf{b} + \boldsymbol{\gamma}_0)] \begin{bmatrix} \hat{\mathbf{x}}_{LS} \\ 1 \end{bmatrix} = 0 \quad (4.3)$$

TLS법(입·출력 모두 관측잡음이 있는 경우)

어떤 행렬 $E \in \mathbf{R}^{m \times (2n+1)}$ 와 벡터 $\boldsymbol{\gamma} \in \mathbf{R}^m$ 에 대해 식(4.4)를 만족하고 $\| [E, \boldsymbol{\gamma}] \|_F$ 를 최소로 하는 E 와 $\boldsymbol{\gamma}$ 을 각각 E_0 와 $\boldsymbol{\gamma}_0$ 라 하면 식(4.5)를 만족하는 $\hat{\mathbf{x}}_{TLS}$ 를 구한다.

$$\mathbf{b} + \boldsymbol{\gamma} \in \text{range}(A + E) \quad (4.4)$$

$$[(A + E_0) \quad -(\mathbf{b} + \boldsymbol{\gamma}_0)] \begin{bmatrix} \hat{\mathbf{x}}_{TLS} \\ 1 \end{bmatrix} = 0 \quad (4.5)$$

출력에만 관측잡음이 있는 시스템을 Fig. 4. 1에 나타냈다.

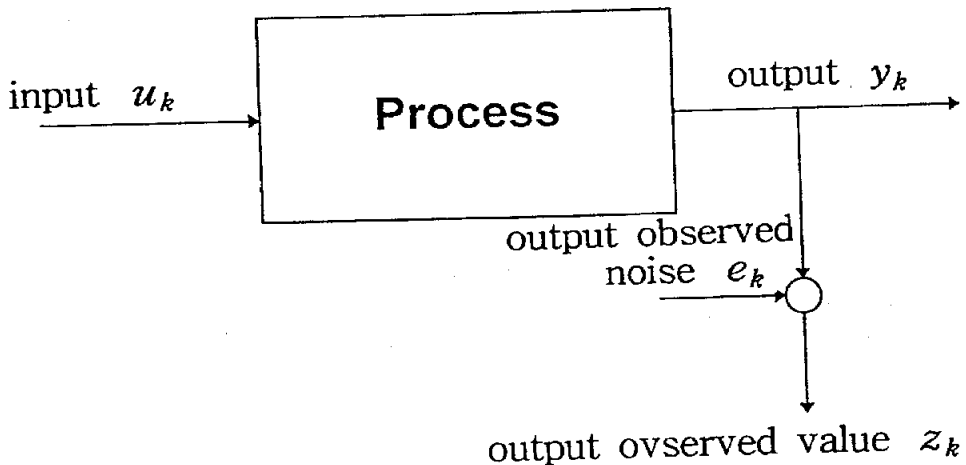


Fig. 4. 1 System model with observed noise in the output

ALS법에서는 입력에 관측잡음이 없는 경우를 고려한다.

식(4.1)의 행렬 A 와 벡터 \mathbf{x} 를 식(4.6)과 같이 분할 할 수 있다.

$$[A_\beta \quad A_\alpha \quad -\mathbf{b}] \begin{bmatrix} \mathbf{x}_\beta \\ \mathbf{x}_\alpha \\ 1 \end{bmatrix} \cong 0 \quad (4.6)$$

$$\text{단, } A_\beta \in \mathbf{R}^{m \times (n+1)}, \quad A_\alpha \in \mathbf{R}^{m \times n}, \quad \mathbf{b} \in \mathbf{R}^m,$$

$$\mathbf{x}_\beta \in \mathbf{R}^{n+1}, \quad \mathbf{x}_\alpha \in \mathbf{R}^n$$

A_β 에는 입력관측치 밖에 들어있지 않기 때문에 즉, 잡음이 없으므로 입력에 관한 파라미터 \mathbf{x}_β 를 LS법으로 계산한다. 한편, A_α 에는 출력관측치가 들어있기 때문에 즉, 잡음이 있으므로 출력에 관한 파라미터 \mathbf{x}_α 를 TLS법에 의해 구한다. 이를 고려하면 ALS법이라 하는 방법을 얻을 수 있다.

ALS법(출력에만 관측잡음이 있는 경우)

어떤 행렬 $E \in \mathbf{R}^{m \times n}$ 과 벡터 $\boldsymbol{\gamma} \in \mathbf{R}^m$ 에 대해 식(4.7)을 만족하고 $\| [E, \boldsymbol{\gamma}] \|_F$ 를 최소로 하는 E 와 $\boldsymbol{\gamma}$ 를 각각 E_0 와 $\boldsymbol{\gamma}_0$ 라 하면 식(4.8)을 만족하는 $\hat{\mathbf{x}}_\alpha$ 와 $\hat{\mathbf{x}}_\beta$ 를 구한다.

$$\mathbf{b} + \boldsymbol{\gamma} \in \text{range}([A_\beta \quad (A_\alpha + E)]) \quad (4.7)$$

$$[A_\beta \quad (A_\alpha + E_0) \quad -(\mathbf{b} + \boldsymbol{\gamma}_0)] \begin{bmatrix} \hat{\mathbf{x}}_\beta \\ \hat{\mathbf{x}}_\alpha \\ 1 \end{bmatrix} = 0 \quad (4.8)$$

4. 2 ALS 추정치

행렬 $[A \quad -\mathbf{b}]$ 의 QR분해를 고려하자.

$$Q^T[A \quad -\mathbf{b}] = \begin{bmatrix} R_A & \boldsymbol{\gamma} \\ 0 & \boldsymbol{\gamma} \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} R \\ 0 \end{bmatrix} \quad (4.9)$$

단, R_A, R 는 상삼각행렬이며, Q 는 직교행렬이다.

직교변환의 성질에 의해 식(4.1)과 식(4.10)은 같은 최소2승해를 가진다.

$$R \begin{bmatrix} \mathbf{x} \\ 1 \end{bmatrix} \cong 0 \quad (4.10)$$

다음은 행렬 R 과 벡터 \mathbf{x} 를 식(4.11)과 같이 분할한다.

$$\begin{bmatrix} R_\beta & W_\alpha \\ 0 & R_\alpha \end{bmatrix} \begin{bmatrix} \mathbf{x}_\beta \\ \mathbf{x}_\alpha \\ 1 \end{bmatrix} \cong 0 \quad (4.11)$$

단, $R_\alpha, R_\beta, W_\alpha \in \mathbb{R}^{(n+1) \times (n+1)}$, $\mathbf{x}_\alpha \in \mathbb{R}^n$, $\mathbf{x}_\beta \in \mathbb{R}^{n+1}$

식(4.11)은 식(4.12), (4.13)과 같이 나눌 수 있다.

$$R_\beta \mathbf{x}_\beta + W_\alpha \begin{bmatrix} \mathbf{x}_\alpha \\ 1 \end{bmatrix} \cong 0 \quad (4.12)$$

$$R_a \begin{bmatrix} \mathbf{x}_a \\ 1 \end{bmatrix} \cong 0 \quad (4.13)$$

식(4.13)의 R_a 를 SVD분해하면 식(4.14)와 같다.

$$U_a^T R_a V_a = \Sigma_a \quad (4.14)$$

$$\text{단, } U_a^T U_a = I, \quad V_a^T V_a = I, \quad \Sigma_a = \text{diag}(\sigma_{a1}, \dots, \sigma_{an}, \sigma_{an+1})$$

$$\sigma_{a1} \geq \dots \geq \sigma_{an} \geq \sigma_{an+1}$$

V_a 를 식(4.15), (4.16)과 같이 분할하면 $\hat{\mathbf{x}}_a$ 는 식(4.17)에 의해 구할 수 있다.

$$V_a = [\mathbf{v}_{a1}, \dots, \mathbf{v}_{an}, \mathbf{v}_{an+1}] = [V_{a1}, \mathbf{v}_{an+1}] \quad (4.15)$$

$$\mathbf{v}_{an+1} = [v_{a1}, \dots, v_{an+1}] \quad (4.16)$$

$$\begin{bmatrix} \hat{\mathbf{x}}_a \\ 1 \end{bmatrix} = \frac{1}{v_{an+1}} \mathbf{v}_{an+1} \quad (4.17)$$

식(4.17)을 식(4.12)에 대입하면 $\hat{\mathbf{x}}_\beta$ 를 계산하는 식이 구해진다.

4. 3 RALS알고리즘

R_k 의 축차계산법을 고려하자.

$$H \begin{bmatrix} R_{k-1} \\ \mathbf{m}_k^T \end{bmatrix} = \begin{bmatrix} R_k \\ 0 \end{bmatrix} = \begin{bmatrix} R_{Ak} & \boldsymbol{\gamma}_k \\ 0 & \gamma \\ 0 & 0 \end{bmatrix} \quad (4.18)$$

가 된다. 단, H 는 직교행렬이다.

RALS법은 다음과 같은 알고리즘이다.

알고리즘 4. 1

행렬 R_{k-1} 과 벡터 \mathbf{m}_k 는 기지라 한다.

1. 식(4.18)에 의해 R_k 를 계산한다.
2. 식(4.19)에 의해 R_{ak} 의 SVD를 계산한다.

$$U_{ak}^T R_{ak} V_{ak} = \begin{bmatrix} \Sigma_{ak} \\ 0 \end{bmatrix} \quad (4.19)$$

$$\text{단, } U_{ak}^T U_{ak} = I, \quad V_{ak}^T V_{ak} = I,$$

$$\Sigma_{ak} = \text{diag}(\sigma_{ak(1)}, \dots, \sigma_{ak(n)}, \sigma_{ak(n+1)})$$

$$\sigma_{ak(1)} \geq \dots \geq \sigma_{ak(n)} \geq \sigma_{ak(n+1)}$$

$$V_{ak} = [\mathbf{v}_{ak(1)}, \dots, \mathbf{v}_{ak(n)}, \mathbf{v}_{ak(n+1)}] = [V_{ak(1)}, \mathbf{v}_{ak(n+1)}]$$

$$\mathbf{v}_{ak(n+1)} = [\mathbf{v}_{ak(1)}, \dots, \mathbf{v}_{ak(n+1)}]^T$$

3. 식(4.20), (4.21)에 의해 파라미터 추정치를 구한다.

$$\begin{bmatrix} \hat{\mathbf{x}}_{ak} \\ 1 \end{bmatrix} = \frac{1}{v_{ak(n+1)}} \mathbf{v}_{ak(n+1)} \quad (4.20)$$

$$\hat{\mathbf{x}}_{\beta k} = -R_{\beta k}^{-1} W_{ak} \begin{bmatrix} \hat{\mathbf{x}}_{ak} \\ 1 \end{bmatrix} \quad (4.21)$$

제5장 컴퓨터 시뮬레이션

본 장에서는 시스템모델을 설정하여 RTLS와 RALS추정법을 이용하여 파라미터 추정과 시뮬레이션 결과를 알아본다.

5. 1 시스템동정

아래의 모델에 대해 RTLS와 RALS추정법을 이용하여 파라미터 추정을 행하였다.

$$G_1(z) = \frac{z^{-1} + 0.5z^{-2}}{1 - 1.5z^{-1} + 0.7z^{-2}}$$

$$G_2(z) = \frac{0.079z^{-1} + 0.047z^{-2}}{1 - 0.975z^{-1} + 0.223z^{-2}}$$

$$G_3(z) = \frac{0.96z^{-1} - 0.48z^{-2} + 0.3z^{-3}}{1 - 1.2z^{-1} + 0.7z^{-2} + 0.1z^{-3}}$$

$$G_4(z) = \frac{0.115z^{-1} + 0.115z^{-3}}{1 - z^{-1} + 0.18z^{-2} - 0.784z^{-3} + 0.656z^{-4}}$$

$$G_5(z) = \frac{1.08z^{-1} - 0.75z^{-2} + 0.45z^{-3} - 0.25z^{-4} + 0.12z^{-5}}{1 - 1.19z^{-1} + 0.81z^{-2} - 0.52z^{-3} + 0.35z^{-4} - 0.12z^{-5}}$$

시스템은 Fig. 5. 1에 나타냈다. 입력신호 $u(k)$ 와 출력관측잡음 $e(k)$ 는 서로 무상관 정규성 백색잡음이며, $u(k) \sim N(0, 1)$, $e(k)$ 의

평균치는 0, 표준편차는 출력신호의 $\frac{1}{10}$ 을 취한다.

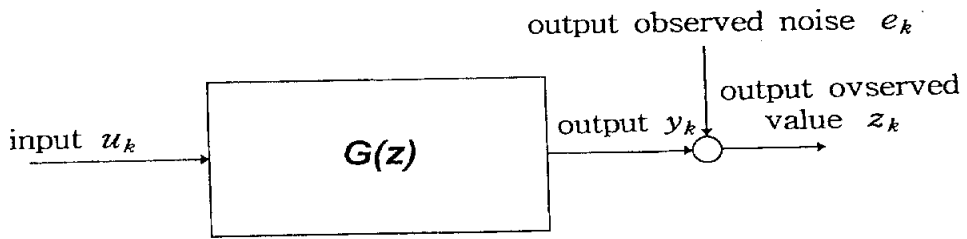


Fig. 5. 1 System model of simulation

각 모델의 시뮬레이션 결과를 다음과 같이 나타냈다.

5. 2 시뮬레이션 결과

$$G_1(z) = \frac{z^{-1} + 0.5z^{-2}}{1 - 1.5z^{-1} + 0.7z^{-2}} \text{의 모델}$$

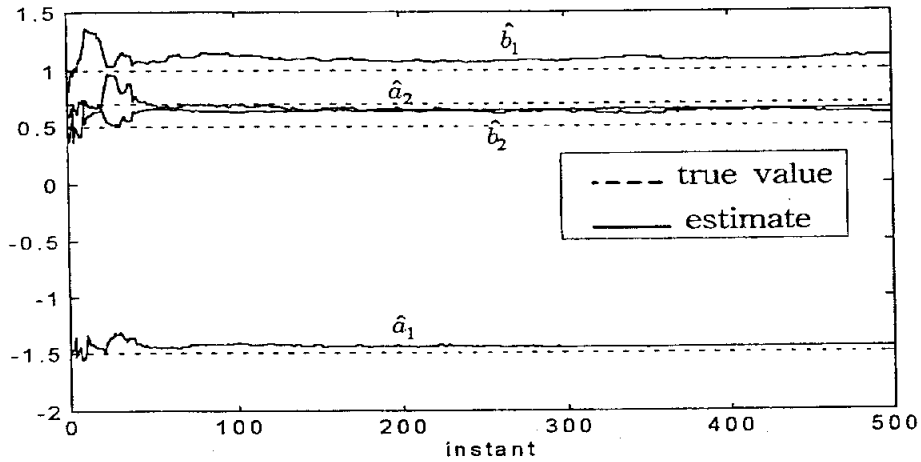


Fig. 5. 2 Parameter estimation by RTLS method

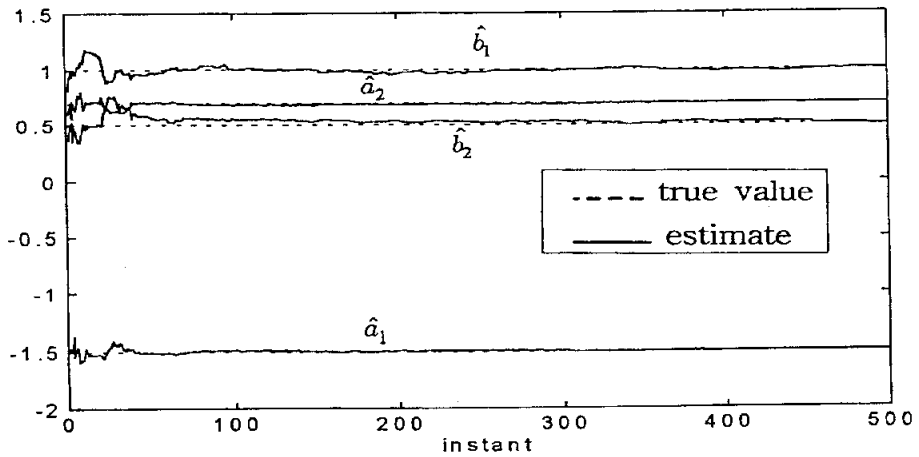


Fig. 5. 3 Parameter estimation by RALS method

Table 1 Parameters and estimates

Parameter	a_1	a_2	b_1	b_2	$y - \hat{y}$ mean value	$y - \hat{y}$ standard deviator
True value	-1.5	0.7	1.0	0.5		
RTLS estimate	-1.4463	-0.6543	1.1077	0.6131	-0.0616	0.3731
RALS estimate	-1.4967	0.6983	1.0090	0.5098	-0.0028	0.0367

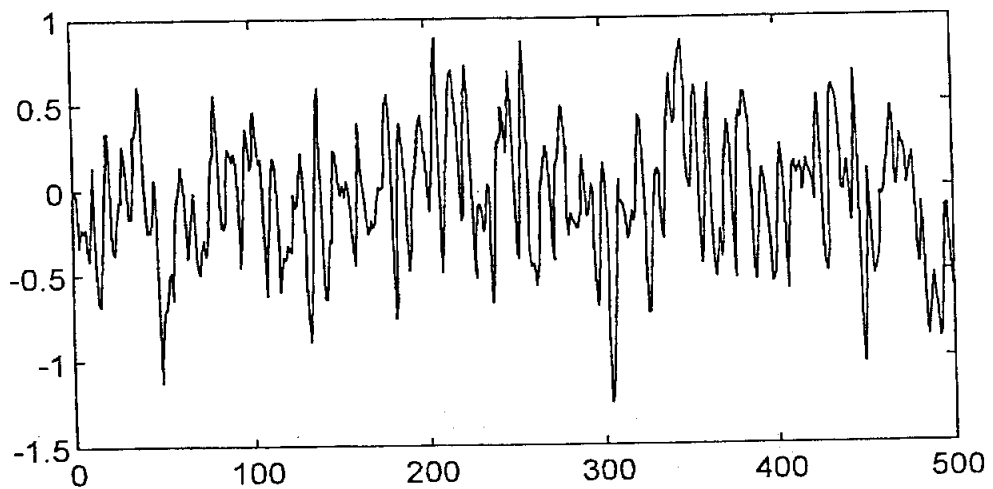


Fig. 5. 4 $y - \hat{y}_{TLS}$

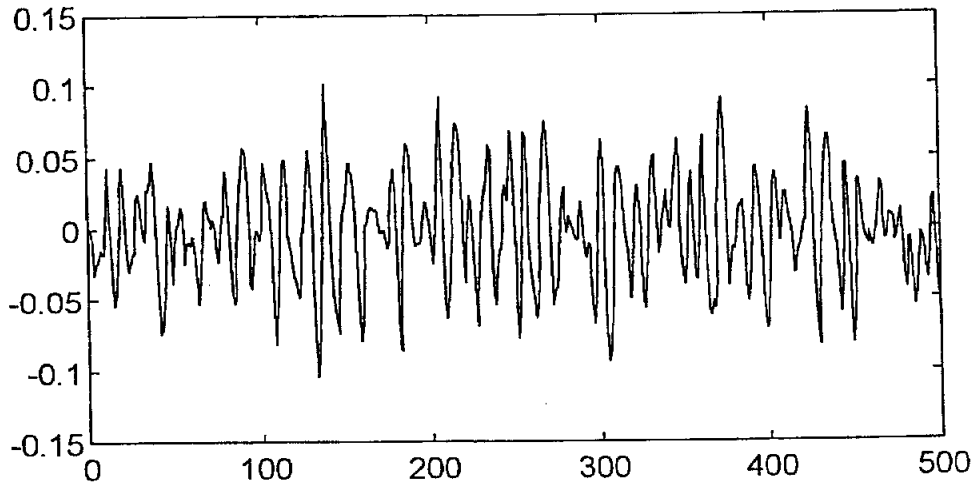


Fig. 5. 5 $y - \hat{y}_{ALS}$

$$G_2(z) = \frac{0.079z^{-1} + 0.047z^{-2}}{1 - 0.975z^{-1} + 0.223z^{-2}} \text{ 의 모델}$$

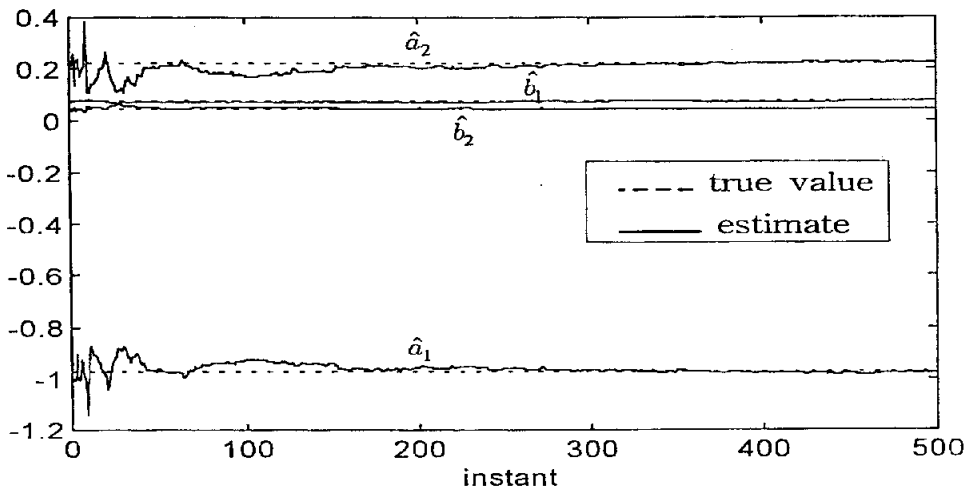


Fig. 5. 6 Parameter estimation by RTLS method

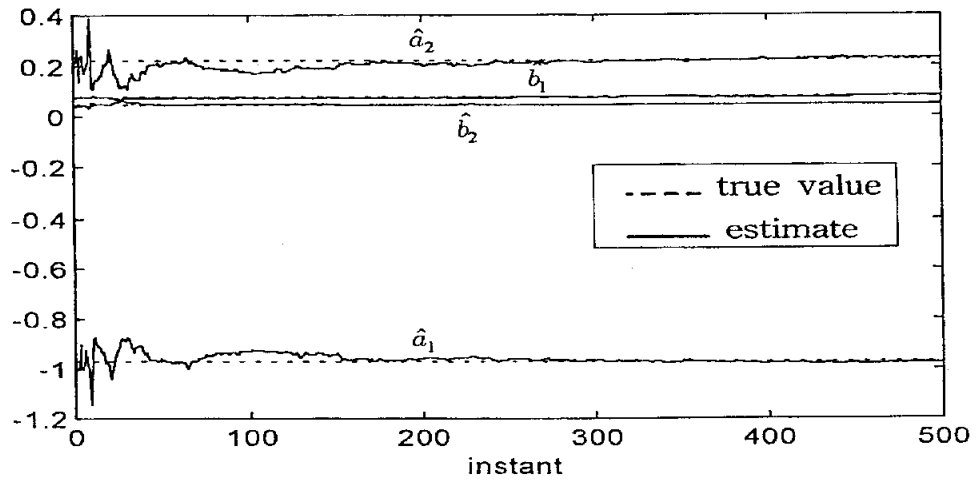


Fig. 5. 7 Parameter estimation by RALS method

Table 2 Parameters and estimates

Parameter	a_1	a_2	b_1	b_2	$y - \hat{y}$ mean value	$y - \hat{y}$ standard deviation
True value	-0.975	0.223	0.079	0.047		
RTLS estimate	-0.9786	0.2265	0.0790	0.0471	-5.7890e -005	8.259e -004
RALS estimate	-0.9726	0.2275	0.0790	0.0470	-3.6468e -005	8.2091e -004

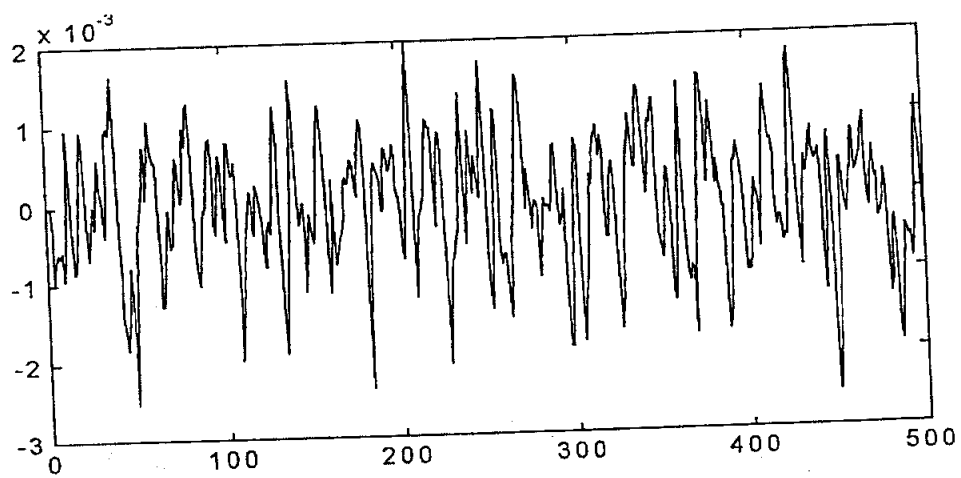


Fig. 5. 8 $y - \hat{y}_{TLS}$

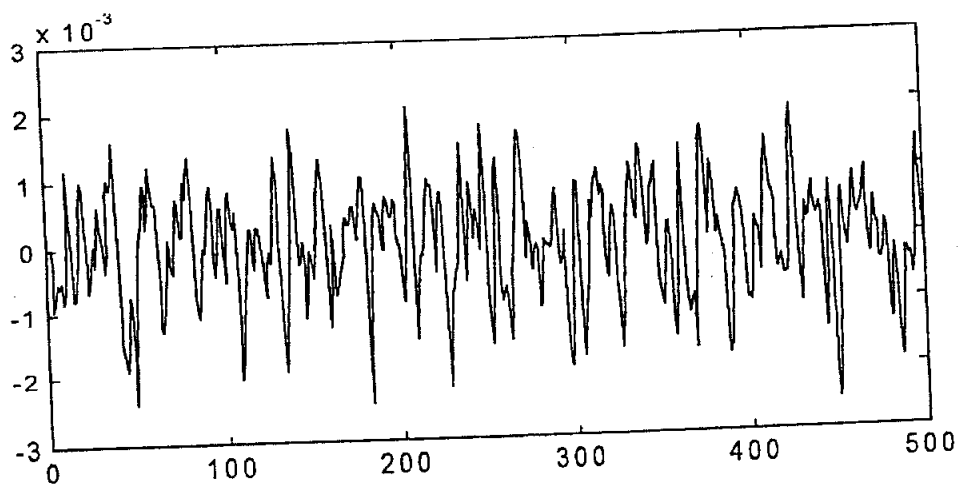


Fig. 5. 9 $y - \hat{y}_{ALS}$

$$G_3(z) = \frac{0.96z^{-1} - 0.48z^{-2} + 0.3z^{-3}}{1 - 1.2z^{-1} + 0.7z^{-2} + 0.1z^{-3}} \text{ 의 모델}$$

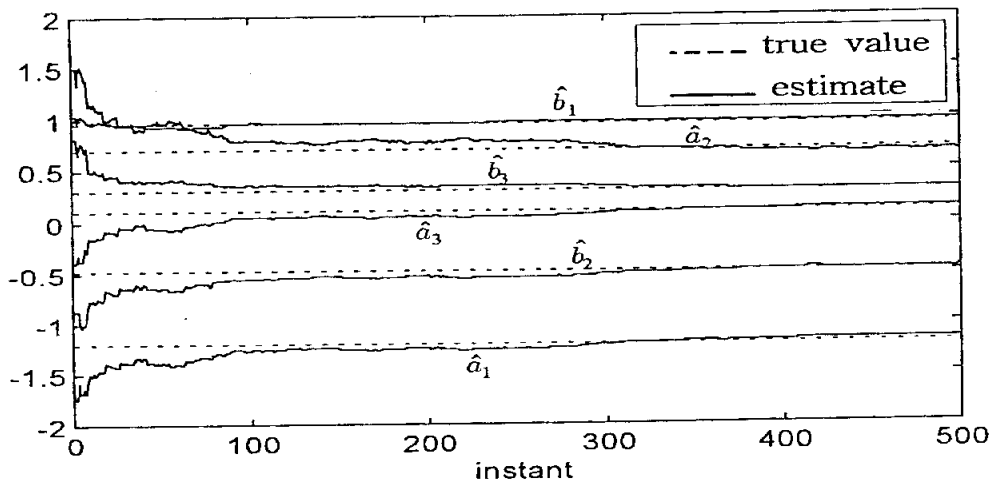


Fig. 5. 10 Parameter estimation by RTLS method

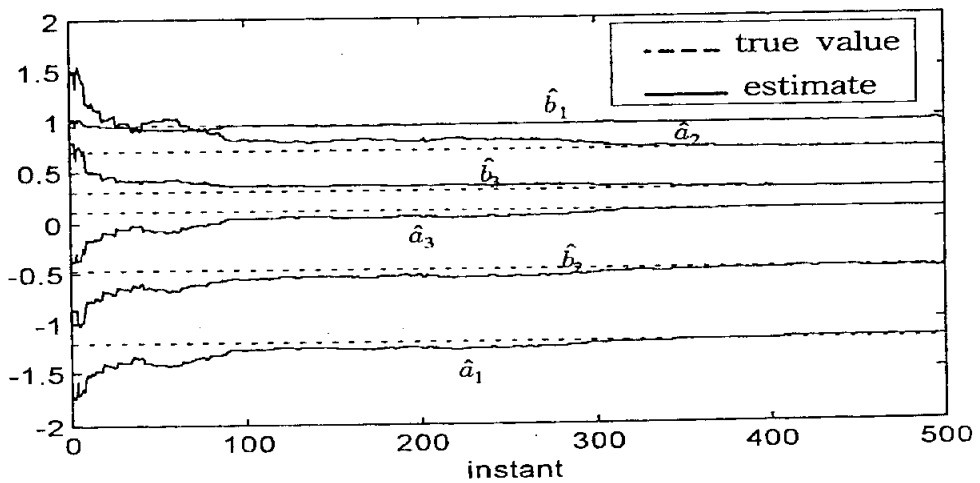


Fig. 5. 11 Parameter estimation by RALS method

Table 3 Parameters and estimates

Parameter	a_1	a_2	a_3	b_1	b_2	b_3
True value	-1.2	0.7	0.1	0.96	-0.48	0.3
RTLS estimate	-1.1709	0.6637	0.1217	0.9728	-0.4682	0.2976
RALS estimate	-1.1857	0.6830	0.1112	0.9664	-0.4795	0.3045

$y - \hat{y}$ mean value	$y - \hat{y}$ standard deviator
3.786e-004	0.0228
4.1827e-005	0.0133

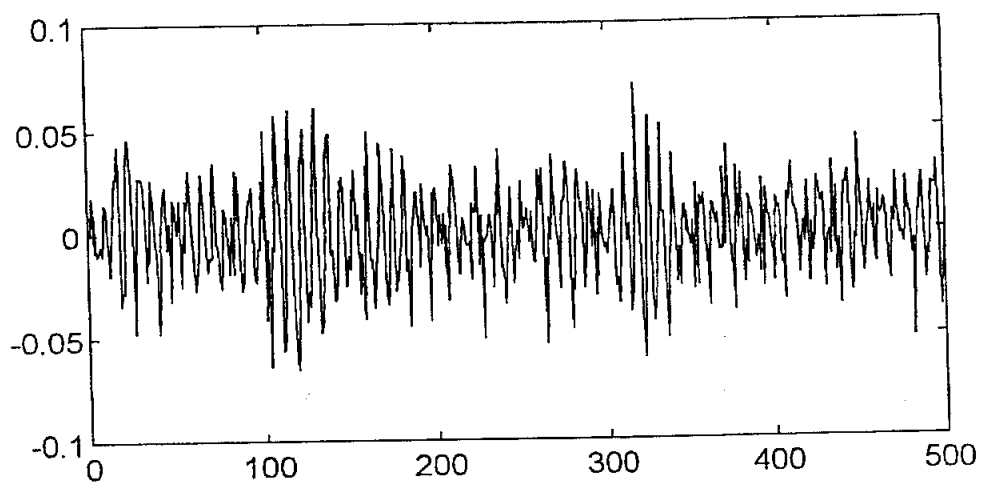


Fig. 5. 12 $y - \hat{y}_{TLS}$

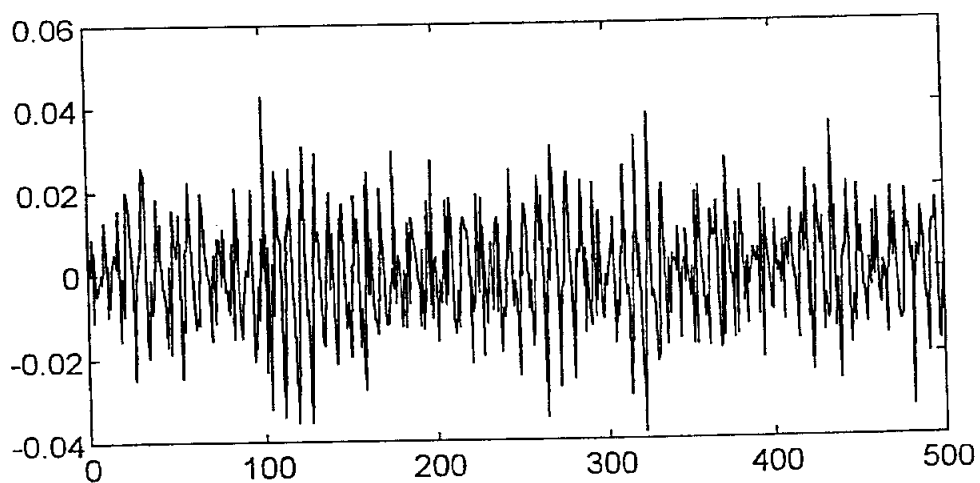


Fig. 5. 13 $y - \hat{y}_{ALS}$

$$G_4(z) = \frac{0.115z^{-1} + 0.115z^{-3}}{1 - z^{-1} + 0.18z^{-2} - 0.784z^{-3} + 0.656z^{-4}} \text{ 의 모델}$$

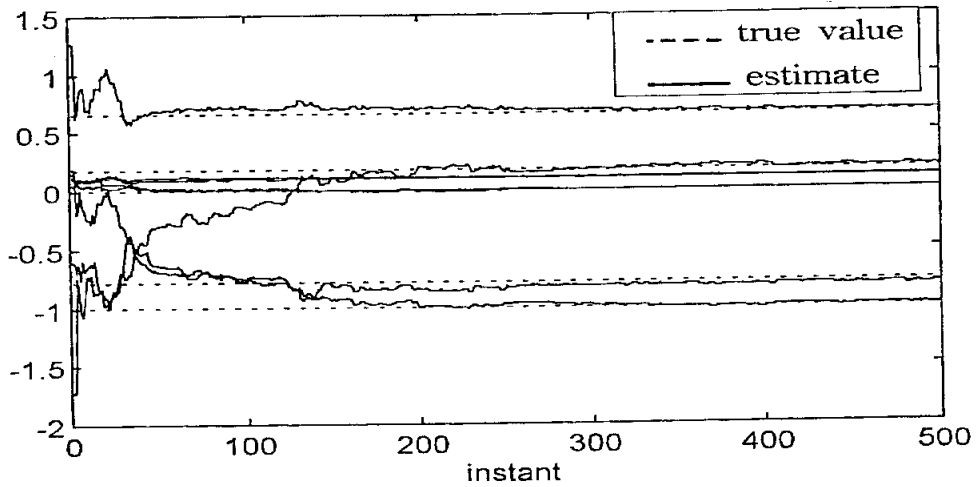


Fig. 5. 14 Parameter estimation by RTLS method

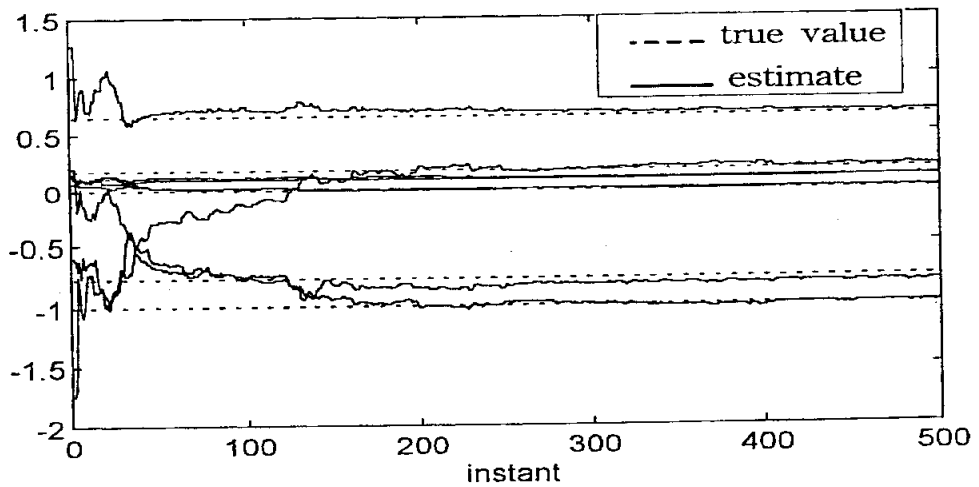


Fig. 5. 15 Parameter estimation by RALS method

Table 4 Parameters and estimates

Parameter	a_1	a_2	a_3	a_4	b_1
True value	-1	0.18	-0.784	0.656	0.115
RTLS estimate	-0.9804	0.1749	-0.8133	0.6724	0.1062
RALS estimate	-0.9839	0.1794	-0.8165	0.646	0.1057

b_2	b_3	b_4	$y - \hat{y}$ mean value	$y - \hat{y}$ standard deviation
0	0.115	0		
0.0049	0.0049	0.0050	1.2127e-004	0.0268
0.0045	0.0045	0.0043	1.9794e-004	0.0293

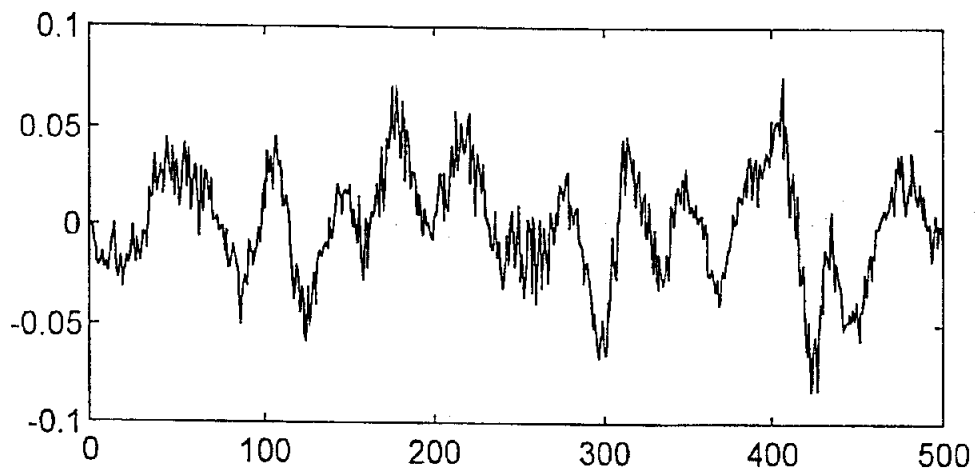


Fig. 5. 16 $y - \hat{y}_{TLS}$

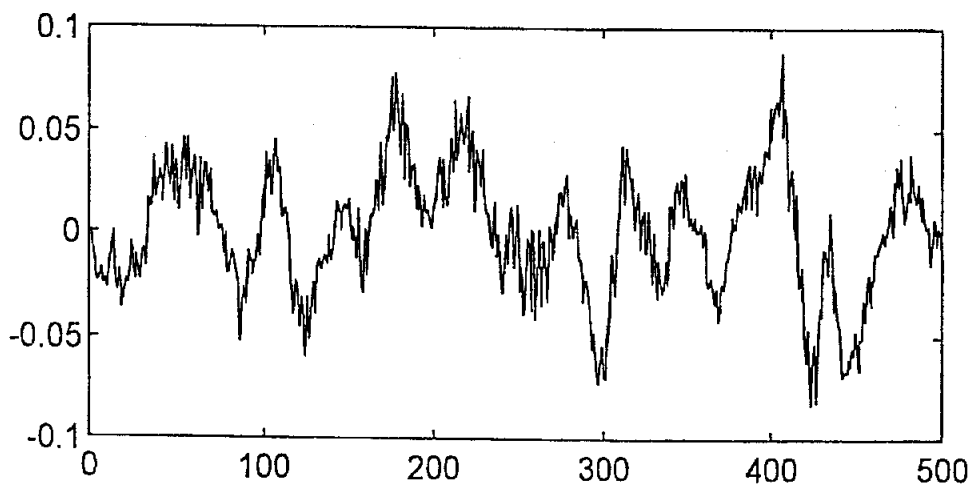


Fig. 5. 17 $y - \hat{y}_{ALS}$

$$G_5(z) = \frac{1.08z^{-1} - 0.75z^{-2} + 0.45z^{-3} - 0.25z^{-4} + 0.12z^{-5}}{1 - 1.19z^{-1} + 0.81z^{-2} - 0.52z^{-3} + 0.35z^{-4} - 0.12z^{-5}}$$

의 모델

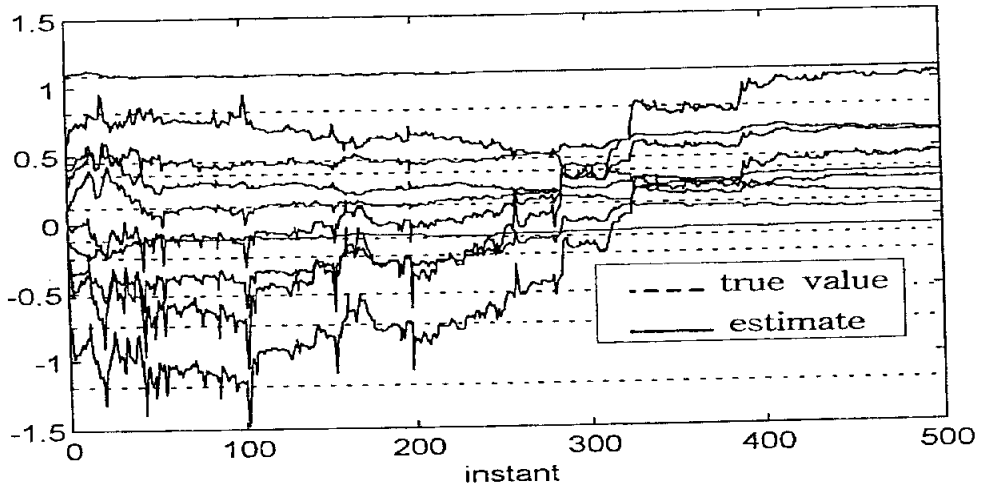


Fig. 5. 18 Parameter estimation by RTLS method

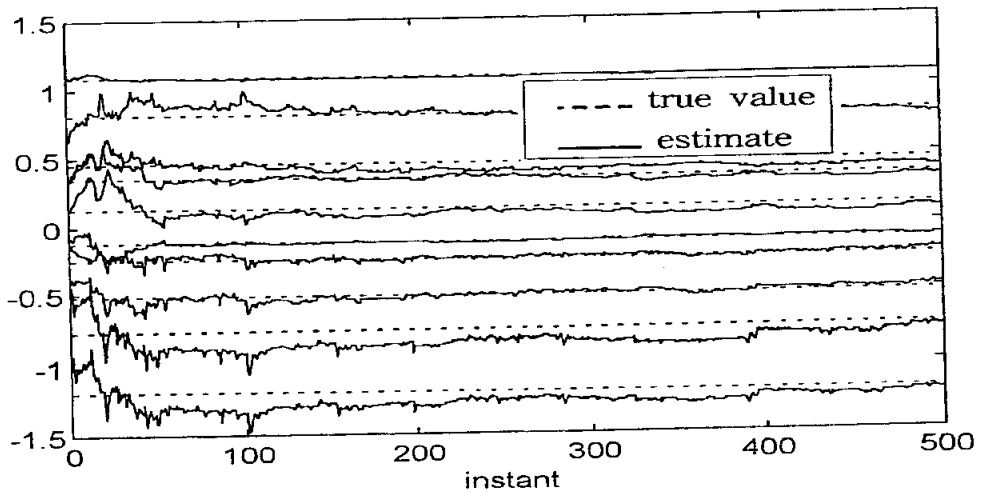


Fig. 5. 19 Parameter estimation by RALS method

Table 5 Parameters and estimates

Parameter	a_1	a_2	a_3	a_4	a_5
True value	-1.19	0.81	-0.52	0.35	-0.12
RTLS estimate	0.4585	0.1587	0.2750	0.0718	-0.0635
RALS estimate	-1.2127	0.7752	-0.4751	0.3204	-0.1164

b_1	b_2	b_3	b_4	b_5	$y - \hat{y}$ mean value	$y - \hat{y}$ standard deviation
1.0816	-0.75	0.45	-0.25	0.12		
1.0806	1.0335	0.6275	0.6138	0.3216		
1.0806	-0.7748	0.3945	-0.2171	0.0927	6.7300e -005	0.0087

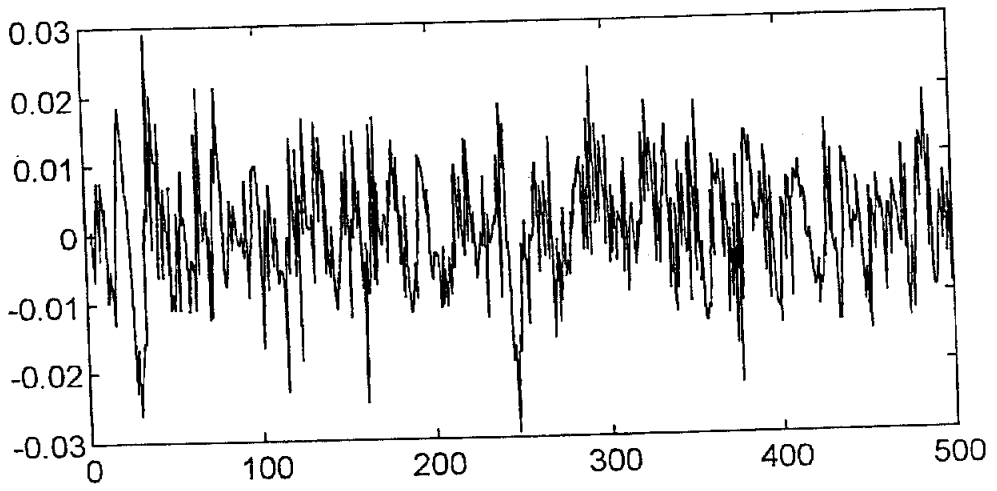


Fig. 5. 20 $y - \hat{y}_{ALS}$

제6장 결 론

본 논문에서는 출력에만 관측잡음이 있는 경우의 일치추정법으로서 LS법과 TLS법의 중간적 방법인 ALS법과 축차알고리즘에 대해 검토했고, 컴퓨터 시뮬레이션에 의해 추정결과를 나타냈다.

잡음이 적을 경우는, TLS법과 ALS법의 추정치가 대개 같지만, 잡음이 클 경우는 ALS법의 추정치는 TLS법의 추정치보다 참값에 근접하고 있다. 특히 5차모델의 경우, TLS법의 추정치보다 ALS법의 추정치가 바이어스가 적고 매우 양호하다.

또 ALS법은 특이치분해를 행하는 행렬이 작기 때문에 응답속도는 TLS법보다 빠르다. 출력에만 관측잡음이 있는 경우, ALS법은 파라미터의 일치추정치가 얻어졌고, 또 TLS법보다 정도(Precision)가 양호한 값이 얻어지므로 유용하다고 사료된다.

참 고 문 헌

- [1] G. C. Goodwin and R. L. Payne, *Theory and Practice of Recursive Identification*, The MIT Press, 1981
- [2] L. Ljung, *System Identification Theory for The User*, Prentice-Hall, 1987
- [3] M. Aoki, *State Space Modeling of Time Series(2nd ed.)*, Springer, 1990
- [4] A. J. Helmicki, C. A. Jacobson, C. N. Nett, "Control-Oriented System Identification : A Worst-Case/Deterministic Approach in H_∞ ," *IEEE Trans. Automat. Contr.*, Vol. AC-39, NO. 10, pp. 1163-1176, 1991
- [5] T. Söderström and P. Stoica, *System Identification*, Prentice-Hall, 1989
- [6] P. Eykhoff and P. C. Parks, "Special Issue on Identification and System Parameter Estimation," *Automatica*, Vol. 26(1), pp. 1-194, 1990
- [7] T. Nakamizo, *Signal Analysis and System Identification*, Corona Publishing Co. LTD., 1988
- [8] T. Uchiyama, J. Imai, K. Wada, "A study on Robust Identification in the Presence of Noise," *JSICE*, Vol. 43, No. 10, pp. 1374-1379, 1998

- [9] Y. Inouye and H. Tsuchiya, Identification of Linear Systems using Input-Output Cummulants, Int. J. Control, Vol. 53 No. 6, pp.1431-1448, 1991
- [10] 吳楊, 片山徹, "バイアス補償法による入出力雑音を受ける線形システムの同定," システム制御情報學會論文誌, Vol. 5, No. 11, pp. 454-460, 1992
- [11] B. Friedlander, "A Modified Prefilter for Some Recursive Estimation Algorithms," IEEE Trans., Automat. Contr., AC-27(1), pp. 232-235, 1982
- [12] 片山徹, システム同定入門, p. 25, 朝倉書店, 1994
- [13] G. H. Golub and C. F. Van Loan, *Matrix Computation*, The Johns Hopkins University Press, U. S. A., 1991
- [14] F. R. Gantmacher, *The Theory of Matrices*, Chelsea Publishing Co., New York, 1960
- [15] 伊理, 兒玉, 須田, 特異値分解とそのシステムへの應用, Vol. 21, No. 8, pp. 763-772, 1982

부록 A

직교변환

[정의] 행렬 $A \in \mathbf{R}^{m \times n}$ 과 $AA^T = A^T A = I$ 가 성립할 때 A 를 직교행렬이라 한다.

A.1 QR분해

[Lemma A.1]

임의의 벡터 $\mathbf{v} \in \mathbf{R}^m$ 에 대해 식(A.1)을 만족하는 직교행렬이 존재한다.

$$Q\mathbf{v} = -\alpha \|\mathbf{v}\|_2 \mathbf{e}_1 \quad (\text{A.1})$$

$$\text{단. } \mathbf{e} = [1 \ 0 \ \cdots \ 0]^T$$

$$\alpha = \begin{cases} +1 & \text{if } v_1 \geq 0 \\ -1 & \text{if } v_1 < 0 \end{cases}$$

[증명]

$\mathbf{v} = 0$ 일 때, 임의의 $m \times m$ 의 직교행렬은 식(A.1)을 만족한다.

$\mathbf{v} \neq 0$ 일 때, 벡터 \mathbf{u} 와 행렬 Q 를

$$\mathbf{u} = \mathbf{v} + \alpha \|\mathbf{v}\|_2 \mathbf{e}_1$$

$$Q = I_m - \frac{2\mathbf{u}\mathbf{u}^T}{\mathbf{u}^T \mathbf{u}}$$

라 두면 Q 는 식(A.1)을 만족하고 직교행렬임을 알 수 있다. ■

[정리A.1] QR분해

임의의 행렬 $A \in \mathbf{R}^{m \times n}$ 에 대해 식(A.2)를 만족하는

$$QA = R \tag{A.2}$$

직교행렬 $Q \in \mathbf{R}^{m \times m}$ 가 존재한다. 단, R 의 주대각선보다 아래의 요소는 0이다.

[증명]

행렬 A 의 제1열에 대하여 [Lemma A.1]을 만족하는 직교행렬 $Q_1 \in \mathbf{R}^{m \times m}$ 을 택하자. 그러면 $Q_1 A$ 의 제1열의 제2에서 제 m 까지의 요소는 0이 된다. 계속하여 $Q_1 A$ 의 제2열의 제2에서 제 m 까지의 요소로 구성된 벡터에 대하여 Q_1 과 같은 직교행렬 $P_2 \in \mathbf{R}^{(m-1) \times (m-1)}$ 을 택하자. 결국 직교행렬 $Q_2 \in \mathbf{R}^{m \times m}$ 을 식(A.3)과 같이 두자.

$$Q_2 = \begin{bmatrix} 1 & 0 \\ 0 & P_2 \end{bmatrix} \tag{A.3}$$

$Q_2 Q_1 A$ 의 주대각선 보다 아래의 제1열과 제2열의 각 요소는 0이 된다. 계속하여 직교행렬 Q_1, \dots, Q_{n-1}, Q_n 는 각각 $m \times m$ 의 직교행렬이므로, $Q = Q_n Q_{n-1} \dots Q_1$ 은 직교행렬이다.

즉, 직교행렬 $Q = Q_n Q_{n-1} \dots Q_1 \in \mathbf{R}^{m \times m}$ 가 존재하고 다음식을 만족한다.

$$QA = R$$



A.2 SVD(특이치분해)

[정리 A.2]

행렬 $A \in \mathbf{R}^{n \times n}$ 에 대해 $\text{rank}(A) = n$, 직교행렬 $U, V \in \mathbf{R}^{n \times n}$ 가 존재하고 식(A.4)를 만족한다.

$$U^T A V = S \text{ or } A = U S V^T \quad (\text{A.4})$$

단, $S = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_n)$

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_n > 0$$

[증명]

$A \in \mathbf{R}^{n \times n}$, $\text{rank}(A) = n$ 보다 $A^T A$ 는 정정대칭행렬(Positive definite symmetric matrix)임을 알 수 있다.

따라서 $A^T A$ 는 식(A.5)와 같은 고유치를 가진다.

$$A^T A = V D V^T \quad (\text{A.5})$$

여기서,

$$V \in \mathbf{R}^{n \times n}, \quad V^T V = I, \quad D = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n),$$

$$\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n > 0$$

I 는 $A^T A$ 의 고유치이다. 행렬을 식(A.6)이라 두면 식(A.7), (A.8)과 같이 나눌 수 있다.

$$S = \text{diag}(\lambda_1^{\frac{1}{2}}, \lambda_2^{\frac{1}{2}}, \dots, \lambda_n^{\frac{1}{2}}) \quad (\text{A.6})$$

$$D = S^T S = S^2 \quad (\text{A.7})$$

$$S^{-1}DS^{-1} = I_n \quad (\text{A.8})$$

다음에 행렬 U 를 식(A.9)와 같다고 하자.

$$U = AVS^{-1} \quad (\text{A.9})$$

$$U^T U = S^{-T} V^T A^T A V S^{-1} = S^{-1} D S^{-1} = I_n \quad (\text{A.10})$$

따라서 행렬 U 는 직교행렬이다. 단 식(A.11)을 만족한다.

$$USV^T = AVS^{-1}SV^T = AVV^T = A \quad (\text{A.11})$$



[Lemma A.2]

임의의 행렬 $A \in \mathbf{R}^{m \times n}$ 에 대해 $\text{rank}(A) = k$, 직교행렬 $H \in \mathbf{R}^{m \times m}$ 와 $K \in \mathbf{R}^{n \times n}$ 가 존재하고 식(A.12)를 만족한다.

$$H^T A K = R \text{ or } A = H R K^T \quad (\text{A.12})$$

$$\text{단, } R = \begin{bmatrix} R_{11} & 0 \\ 0 & 0 \end{bmatrix} \quad (\text{A.13})$$

R_{11} 은 정칙 상삼각행렬이므로 $R_{11} \in \mathbf{R}^{k \times k}$ 이다.

[증명]

(생략)

[정리 A.3] 특이치분해

임의의 행렬 $A \in \mathbf{R}^{m \times n}$ 에 대해 $rank(A) = k$, 직교행렬 $U \in \mathbf{R}^{m \times m}$ 와 $V \in \mathbf{R}^{n \times n}$ 가 존재하고 식(A.14)를 만족한다.

$$U^T A V = S \text{ or } A = U S V^T \quad (\text{A.14})$$

단, $S \in \mathbf{R}^{m \times n}$

$$S = \begin{bmatrix} S_{11} & 0 \\ 0 & 0 \end{bmatrix},$$

$$S_{11} = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_k)$$

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_k > 0 \quad (\text{A.15})$$

[증명]

Lemma A.2에 의해 직교행렬 $H \in \mathbf{R}^{m \times m}$ 와 $K \in \mathbf{R}^{n \times n}$ 가 존재하고 식(A.16)을 만족한다.

$$A = H R K^T = H \begin{bmatrix} R_{11} & 0 \\ 0 & 0 \end{bmatrix} K^T \quad (\text{A.16})$$

단, R_{11} 은 정칙 상삼각행렬이며 $R_{11} \in \mathbf{R}^{k \times k}$ 이다.

정리 A.2에 의해 직교행렬 $\tilde{U}, \tilde{V} \in \mathbf{R}^{k \times k}$ 가 존재하고 식(A.17)을 만족한다.

$$R_{11} = \tilde{U} S_{11} \tilde{V}^T \quad (\text{A.17})$$

단, $S_{11} = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_k)$

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_k > 0$$

여기서 행렬을 식(A.18), (A.19)라 두면 식(A.20), (A.21)이 성립함

을 알 수 있다.

즉, \hat{U} 와 \hat{V} 는 직교행렬이다.

$$\hat{U} = \begin{bmatrix} \hat{U} & 0 \\ 0 & I_{m-k} \end{bmatrix} \quad (\text{A.18})$$

$$\hat{V} = \begin{bmatrix} \hat{V} & 0 \\ 0 & I_{n-k} \end{bmatrix} \quad (\text{A.19})$$

$$\hat{U}\hat{U}^T = \hat{U}^T\hat{U} = I_m \quad (\text{A.20})$$

$$\hat{V}\hat{V}^T = \hat{V}^T\hat{V} = I_n \quad (\text{A.21})$$

또 행렬 $S \in \mathbf{R}^{m \times n}$ 을 식(A.22)라 두면 식(A.23)이 된다.

$$S = \begin{bmatrix} S_{11} & 0 \\ 0 & 0 \end{bmatrix} \quad (\text{A.22})$$

$$\begin{aligned} \hat{U}S\hat{V}^T &= \begin{bmatrix} \hat{U} & 0 \\ 0 & I_{m-k} \end{bmatrix} \begin{bmatrix} S_{11} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{V} & 0 \\ 0 & I_{n-k} \end{bmatrix}^T \\ &= \begin{bmatrix} \hat{U}S_{11}\hat{V}^T & 0 \\ 0 & 0 \end{bmatrix}_{m \times n} \\ &= \begin{bmatrix} R_{11} & 0 \\ 0 & 0 \end{bmatrix}_{m \times n} \\ &= R \end{aligned} \quad (\text{A.23})$$

또, 행렬을 식(A.24), (A.25)라 두면 식(A.26), (A.27), (A.28)이 된다.

$$U = H\hat{U} \quad (\text{A.24})$$

$$V = K\hat{V} \quad (\text{A.25})$$

$$USV^T = H\hat{U}S(K\hat{V})^T = H\hat{U}S\hat{V}^TK^T$$

$$= HRK^T = A \quad (\text{A.26})$$

또,

$$UU^T = H\hat{U}(H\hat{U})^T = H\hat{U}\hat{U}^T H^T = HH^T = I_m \quad (\text{A.27})$$

$$VV^T = K\hat{V}(K\hat{V})^T = K\hat{V}\hat{V}^T K^T = KK^T = I_n \quad (\text{A.28})$$

그러므로 U 와 V 는 직교행렬이다. ■

A Study on the ALS Method of System Identification

Jee-Youl Lee

*Dept. of Electrical Engineering,
Graduate School
Pukyong National University*

Abstract

A system identification is to measure the output in the presence of adequate input for the controlled system and to estimate the mathematical model on the basis of input-output data.

In the traditional system control field, most identification problems have been thought as estimating the unknown modeling parameters on the assumption that the model structures are fixed.

In the system identification, it is possible to estimate the true parameter values by the adjusted least squares method in the input-output case of no observed noise, and it is possible to estimate the true parameter values by the total least squares method in the input-output case with the observed noise. Recently the adjusted least squares method is suggested as a consistent estimation method in the system identification in the case where there is observed noise only in the output.

In this paper the adjusted least squares method has been developed from the least squares method and the efficiency of the estimating results was confirmed by the generating data with the computer simulations.

감사의 글

길게 느껴졌던 학위과정이 벌써 논문 완성이라는 마무리의 시간이 되었습니다. 항상 부족한 저에게 연구의 길을 열어주시고 본 논문이 완성되기까지 아낌없이 가르침을 주신 지도교수 이동철 교수님께 감사드립니다. 아울러 전기공학과 교수님들 정말 고맙습니다.

논문이 완성되기까지 밤낮으로 귀중한 시간을 함께한 연구실 동기 정익에게 진심으로 고마운 마음을 전합니다. 전기공학과와 모든 동기들에게도 감사드립니다.

지금의 순간이 있게 한 모든 분들에게 깊은 감사를 드리며 늘 든든하게 힘이 되어주신 부모님께 감사드립니다.

2002년 12월

이 지 열 올림