Summary of SI

BY WK.H

HITSZ
vvkc@foxmail.com
November 9, 2019

1 Introduction

1.1 Comparison of two mathematical modeling methods

i. Mechanism modeling is to write down a complete set of equations relating the different variables in the system based on physical laws.

ii. System identification (SI) is to find out the relationship between input and output, as well as other signals, from a set of experimental data.

1.2 Definition of SI

System identification can be defined as the determination of a mathematic model from the observed input and output by minimizing some error criterion function.

1.3 Four entities of SI

Data, set of models, criterion, optimization approaches.

2 Identification models and LS estimation

2.1 Identification models

Shift operator

i. Forward z : $zu(t) = u(t+1)$

ii. Backward z^{-1} : $z^{-1}u(t) = u(t-1)$

2.1.1 Time series models

Denote

$$\begin{aligned} A(z^{-1}) &= 1 + \sum_{i=1}^{n_a} a_i z^{-i} \\ B(z^{-1}) &= \sum_{i=1}^{n_b} b_i z^{-i} \\ C(z^{-1}) &= 1 + \sum_{i=1}^{n_c} c_i z^{-i} \\ D(z^{-1}) &= 1 + \sum_{i=1}^{n_d} d_i z^{-i} \end{aligned}$$

i. Autoregressive (AR) model

$$A(z^{-1})y(t) = v(t)$$

ii. Moving average (MA) model

$$y(t) = D(z^{-1})v(t)$$

iii. Autoregressive moving average (ARMA) model

$$A(z^{-1})y(t) = D(z^{-1})v(t)$$

2.1.2 Equation error type models

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

$$w(t) = \begin{cases} v(t) & \text{ARX} \\ D(z^{-1})v(t) & \text{ARMAX} \\ \frac{1}{C(z^{-1})}v(t) & \text{ARARX} \\ \frac{D(z^{-1})}{C(z^{-1})}v(t) & \text{ARARMAX} \end{cases}$$

where AR refers to the autoregressive part $A(z^{-1})y(t)$ and X refers to the extra input $B(z^{-1})u(t)$.

2.1.3 Output error type models

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

$$w(t) = \begin{cases} v(t) & \text{OE} \\ D(z^{-1})v(t) & \text{OEMA} \\ \frac{1}{C(z^{-1})}v(t) & \text{OEAR} \\ \frac{D(z^{-1})}{C(z^{-1})}v(t) & \text{OEARMA}\,(B-J) \end{cases}$$

where $\frac{B(z^{-1})}{A(z^{-1})} := x(t)$ can be viewed as the true output.

2.2 LS principle

2.2.1 One-dimensional case

Question 1. For a desk, we assume that n persons obtain n different values of the length of this desk $x_1, x_2, ..., x_n$. Now our question is, what is the most likely length of this desk?

Answer 1. Intuitively, if x is the length of this desk, it should minimize the following error function

$$f(x) = \sum_{i=1}^{n} (x_i - x)^2,$$

which is the sum of squares of the error. By setting the derivation to zero, one has

$$\frac{df(x)}{dx} = -2\sum_{i=1}^{n} (x_i - x) = 0,$$

which gives

$$x = \frac{x_1 + x_2 + \cdots + x_n}{n}.$$

In addition, it is easily obtained that

$$\frac{df(x)}{dx} = 2n > 0.$$

Therefore, $f\left(\frac{x_1 + x_2 + \cdots + x_n}{n}\right)$ is the minimum value of the error function $f(x)$. This is the basic principle of the method of least squares.

2.3 Statistical properties for LS estimation

Theorem 2.1. *Consider the system (*), and suppose that V_t is of mean zero, and V_t and H_t are statistically independent. Then the least squares estimate $\hat{\theta}_{\mathrm{LS}} = (H_t^T H_t)^{-1} H_t^T Y_t$ is an **unbiased** estimate, that is, $\mathbb{E}\left[\hat{\theta}_{\mathrm{LS}}\right] = \theta$.*

Proof. *By using $Y_t = H_t\theta + V_t$, one has*

$$\begin{aligned}
\hat{\theta}_{\mathrm{LS}} &= (H_t^T H_t)^{-1} H_t^T Y_t \\
&= (H_t^T H_t)^{-1} H_t^T (H_t\theta + V_t) \\
&= (H_t^T H_t)^{-1} H_t^T H_t\theta + (H_t^T H_t)^{-1} H_t^T V_t \\
&= \theta + (H_t^T H_t)^{-1} H_t^T V_t
\end{aligned}$$

then

$$\mathbb{E}\left[\hat{\theta}_{\mathrm{LS}}\right] = \theta + \mathbb{E}[(H_t^T H_t)^{-1} H_t^T V_t] = \theta$$

□

Theorem 2.2. *Consider the system (*), and suppose that V_t is of mean zero and has the covariance matrix $\mathrm{cov}[V_t] = R_v$. In addition, it is assumed that V_t and H_t are statistically independent. Then the covariance matrix of the estimation error $\tilde{\theta}_{\mathrm{LS}}(t) = \hat{\theta}_{\mathrm{LS}}(t) - \theta$ is given by*

$$\mathrm{cov}\left[\tilde{\theta}_{\mathrm{LS}}(t)\right] = \mathbb{E}[(H_t^T H_t)^{-1} H_t^T R_v H_t (H_t^T H_t)^{-1}]$$

Proof. *It follows from the proof of the previous theorem that*

$$\tilde{\theta}_{\mathrm{LS}} = \hat{\theta}_{\mathrm{LS}}(t) - \theta = (H_t^T H_t)^{-1} H_t^T V_t$$

and

$$\mathbb{E}\left[\tilde{\theta}_{\mathrm{LS}}\right] = \mathbb{E}\left[\hat{\theta}_{\mathrm{LS}}(t) - \theta\right] = 0$$

then

$$\begin{aligned}
\mathrm{cov}\left[\tilde{\theta}_{\mathrm{LS}}\right] &= \mathbb{E}\left[\tilde{\theta}_{\mathrm{LS}} \tilde{\theta}_{\mathrm{LS}}^T\right] \\
&= \mathbb{E}[(H_t^T H_t)^{-1} H_t^T V_t V_t^T H_t (H_t^T H_t)^{-1}] \\
&= \mathbb{E}[(H_t^T H_t)^{-1} H_t^T R_v H_t (H_t^T H_t)^{-1}]
\end{aligned}$$

□

If $\mathbb{E}[v(t)] = 0$ and $E[v^2(t)] = \sigma^2$, one has $\mathrm{cov}[V_t] = \sigma^2 I_t$, $\mathrm{cov}\big[\tilde{\theta}_{\mathrm{LS}}(t)\big] = \sigma^2 \mathbb{E}[(H_t^T H_t)^{-1}]$.

Theorem 2.3. *An unbiased estimate of σ^2 can be given by*

$$\hat{\sigma}^2 = \frac{J\big(\hat{\theta}_{\mathrm{LS}}(t)\big)}{t - \dim\theta}$$

3 RLS

3.1 Basic RLS

3.1.1 $P(0)$

When the initial value $P(0)$ takes $p_0 I > 0$, it follows that

$$\begin{aligned} P(t) &= \left[\frac{1}{p_0} I + \sum_{i=1}^{t} \varphi(i)\varphi^T(i)\right]^{-1} \\ &= \left[\frac{1}{p_0} I + H_t^T H_t\right]^{-1} \end{aligned}$$

where

$$H_t = [\ \varphi(1) \ \ \varphi(2) \ \ \dots \ \ \varphi(t)\ ]^T$$

This implies that

$$\begin{aligned} \lim_{p_0 \to \infty} P(t) &= \lim_{p_0 \to \infty} \left[\frac{1}{p_0} I + H_t^T H_t\right]^{-1} \\ &= [H_t^T H_t]^{-1} \\ &= P(t) \end{aligned}$$

This fact shows that the p_0 should be chosen to be as large as possible.

3.2 Forgetting factor RLS algorithm

$$\begin{aligned} J(\theta) &= \sum_{i=1}^{t} \lambda^{t-i} [y(i) - \varphi^T(i)\theta]^2 \\ &= (Y_t - H_t\theta)^T \Lambda_t (Y_t - H_t\theta) \end{aligned}$$

where

$$\begin{aligned} \lambda &= \rho^2 \\ H_t &= [\ \varphi(1) \ \ \varphi(2) \ \ \dots \ \ \varphi(t-1) \ \ \varphi(t)\ ]^T \\ Y_t &= [\ y(1) \ \ y(2) \ \ \dots \ \ y(t-1) \ \ y(t)\ ]^T \\ \Lambda_t &= \mathrm{diag}\{\lambda^{t-1}, \lambda^{t-2}, \dots, \lambda, 1\} \end{aligned}$$

Take the derivation,

$$\begin{aligned}\frac{\partial J(\theta)}{\partial\theta} &= \frac{\partial V_t}{\partial\theta}\frac{\partial J(\theta)}{\partial V_t}\\ &= \frac{\partial(Y_t-H_t\theta)}{\partial\theta}\frac{\partial(V_t^T\Lambda_tV_t)}{\partial V_t}\\ &= -2H_t^T\Lambda_tV_t\\ &= -2H_t^T\Lambda_t(Y_t-H_t\theta)\\ &= 0\end{aligned}$$

which gives

$$\hat{\theta}(t)=(H_t^T\Lambda_tH_t)^{-1}H_t^T\Lambda_tY_t$$

Denote $P(t)=(H_t^T\Lambda_tH_t)^{-1}$, then

$$\begin{aligned}P^{-1}(t) &= \sum_{i=1}^{t}\lambda^{t-i}\varphi(i)\varphi^T(i)\\ &= \lambda\sum_{i=1}^{t-1}\lambda^{t-i-1}\varphi(i)\varphi^T(i)+\varphi(t)\varphi^T(t)\\ &= \lambda P^{-1}(t-1)+\varphi(t)\varphi^T(t)\end{aligned}$$

By the matrix inversion lemma, one has

$$P(t) = \frac{1}{\lambda}\left(P(t-1)-\frac{P(t-1)\varphi(t)\varphi^T(t)P(t-1)}{\lambda+\varphi^T(t)P(t-1)\varphi(t)}\right)$$

$$\begin{aligned}H_t^T\Lambda_tY_t &= \sum_{i=1}^{t}\lambda^{t-i}\varphi(i)y(i)\\ &= \lambda\sum_{i=1}^{t-1}\lambda^{t-i-1}\varphi(i)y(i)+\varphi(t)y(t)\\ &= \lambda H_{t-1}^T\Lambda_{t-1}Y_{t-1}+\varphi(t)y(t)\end{aligned}$$

$$\begin{aligned}\hat{\theta}(t) &= (H_t^T\Lambda_tH_t)^{-1}H_t^T\Lambda_tY_t\\ &= P(t)P(t-1)^{-1}P(t-1)(\lambda H_{t-1}^T\Lambda_{t-1}Y_{t-1}+\varphi(t)y(t))\\ &= \lambda P(t)P(t-1)^{-1}\hat{\theta}(t-1)+P(t)\varphi(t)y(t)\\ &= [I-P(t)\varphi(t)\varphi^T(t)]\hat{\theta}(t-1)+P(t)\varphi(t)y(t)\\ &= \hat{\theta}(t-1)+P(t)\varphi(t)[y(t)-\varphi^T(t)\hat{\theta}(t-1)]\end{aligned}$$

$$\begin{aligned}L(t) &= P(t)\varphi(t)=\frac{P(t-1)\varphi(t)}{\lambda+\varphi^T(t)P(t-1)\varphi(t)}\\ P(t) &= \frac{1}{\lambda}(I-L(t)\varphi^T(t))P(t-1)\end{aligned}$$

3.3 Fixed memory identification

$$\begin{aligned}J(\theta) &= \sum_{i=t-p+1}^{t}[y(i)-\varphi^T(i)\theta]^2\\ &= (Y_{p,t}-H_{p,t}\theta)^T(Y_{p,t}-H_{p,t}\theta)\end{aligned}$$

then

$$\hat{\theta}(t) = (H_{p,t}^T H_{p,t})^{-1} H_{p,t}^T Y_{p,t}$$

where

$$\begin{aligned} H_{p,t} &= [\ \varphi(t-p+1) \ \ \varphi(t-p+2) \ \ \ldots \ \ \varphi(t)\ ]^T \\ Y_{p,t} &= [\ y(t-p+1) \ \ y(t-p+2) \ \ \ldots \ \ y(t)\ ]^T \end{aligned}$$

Denote $P(t) = (H_{p,t}^T H_{p,t})^{-1}$, thus

$$\begin{aligned} P^{-1}(t) &= \sum_{i=t-p+1}^{t} \varphi(i)\varphi^T(i) \\ &= \sum_{i=t-p}^{t-1} \varphi(i)\varphi^T(i) + \varphi(t)\varphi^T(t) - \varphi(t-p)\varphi^T(t-p) \\ &= P^{-1}(t-1) + \varphi(t)\varphi^T(t) - \varphi(t-p)\varphi^T(t-p) \end{aligned}$$

$$\begin{aligned} H_{p,t}^T Y_{p,t} &= \sum_{i=t-p+1}^{t} \varphi(i)y(i) \\ &= \sum_{i=t-p}^{t-1} \varphi(i)y(i) + \varphi(t)y(t) - \varphi(t-p)y(t-p) \\ &= H_{p,t-1}^T Y_{p,t-1} + \varphi(t)y(t) - \varphi(t-p)y(t-p) \end{aligned}$$

so

$$\begin{aligned} \hat{\theta}(t) &= P(t)P^{-1}(t-1)P(t-1)[H_{p,t-1}^T Y_{p,t-1} + \varphi(t)y(t) - \varphi(t-p)y(t-p)] \\ &= P(t)P^{-1}(t-1)\hat{\theta}(t-1) + P(t)[\varphi(t)y(t) - \varphi(t-p)y(t-p)] \\ &= [I - P(t)\varphi(t)\varphi^T(t) + P(t)\varphi(t-p)\varphi^T(t-p)]\hat{\theta}(t-1) + P(t)[\varphi(t)y(t) - \varphi(t-p)y(t-p)] \\ &= \hat{\theta}(t-1) + P(t)\varphi(t)[y(t) - \varphi^T(t)\hat{\theta}(t-1)] - P(t)\varphi(t-p)[y(t-p) - \varphi^T(t-p)\hat{\theta}(t-1)] \\ &= \hat{\theta}(t-1) + P(t)[\ \varphi(t) \ \ -\varphi(t-p)\ ] \begin{bmatrix} y(t) - \varphi^T(t)\hat{\theta}(t-1) \\ y(t-p) - \varphi^T(t-p)\hat{\theta}(t-1) \end{bmatrix} \end{aligned}$$

3.4 Fixed memory identification with a forgetting factor

$$\begin{aligned} J(\theta) &= \sum_{i=t-p+1}^{t} \lambda^{t-i}[y(i) - \varphi^T(i)\theta]^2 \\ &= (Y_{p,t} - H_{p,t}\theta)^T \Lambda (Y_{p,t} - H_{p,t}\theta) \end{aligned}$$

then

$$\hat{\theta}(t) = (H_{p,t}^T \Lambda H_{p,t})^{-1} H_{p,t}^T \Lambda Y_{p,t}$$

where

$$\begin{aligned} H_{p,t} &= [\ \varphi(t-p+1) \ \ \varphi(t-p+2) \ \ \ldots \ \ \varphi(t)\ ]^T \\ Y_{p,t} &= [\ y(t-p+1) \ \ y(t-p+2) \ \ \ldots \ \ y(t)\ ]^T \\ \Lambda &= \mathrm{diag}\{\lambda^{p-1}, \lambda^{p-2}, \ldots, \lambda, 1\} \end{aligned}$$

Denote $P(t) = (H_{p,t}^T \Lambda H_{p,t})^{-1}$, thus

$$\begin{aligned} P^{-1}(t) &= \sum_{i=t-p+1}^{t} \lambda^{t-i}\varphi(i)\varphi^T(i) \\ &= \sum_{i=t-p}^{t-1} \lambda^{t-i}\varphi(i)\varphi^T(i) + \varphi(t)\varphi^T(t) - \lambda^p\varphi(t-p)\varphi^T(t-p) \\ &= P^{-1}(t-1) + \varphi(t)\varphi^T(t) - \lambda^p\varphi(t-p)\varphi^T(t-p) \end{aligned}$$

$$\begin{aligned} H_{p,t}^T \Lambda Y_{p,t} &= \sum_{i=t-p+1}^{t} \lambda^{t-i}\varphi(i)y(i) \\ &= \sum_{i=t-p}^{t-1} \lambda^{t-i}\varphi(i)y(i) + \varphi(t)y(t) - \lambda^p\varphi(t-p)y(t-p) \\ &= H_{p,t-1}^T \Lambda Y_{p,t-1} + \varphi(t)y(t) - \lambda^p\varphi(t-p)y(t-p) \end{aligned}$$

so

$$\begin{aligned} \hat{\theta}(t) &= P(t)P^{-1}(t-1)P(t-1)[H_{p,t-1}^T \Lambda Y_{p,t-1} + \varphi(t)y(t) - \lambda^p\varphi(t-p)y(t-p)] \\ &= P(t)P^{-1}(t-1)\hat{\theta}(t-1) + P(t)[\varphi(t)y(t) - \lambda^p\varphi(t-p)y(t-p)] \\ &= [I - P(t)\varphi(t)\varphi^T(t) + \lambda^p P(t)\varphi(t-p)\varphi^T(t-p)]\hat{\theta}(t-1) + P(t)[\varphi(t)y(t) - \lambda^p\varphi(t-p)y(t-p)] \\ &= \hat{\theta}(t-1) + P(t)\{\varphi(t)[y(t) - \varphi^T(t)\hat{\theta}(t-1)] - \lambda^p\varphi(t-p)[y(t-p) - \varphi^T(t-p)\hat{\theta}(t-1)]\} \\ &= \hat{\theta}(t-1) + P(t)[\ \varphi(t) \quad -\lambda^p\varphi(t-p)\ ]\begin{bmatrix} y(t) - \varphi^T(t)\hat{\theta}(t-1) \\ y(t-p) - \varphi^T(t-p)\hat{\theta}(t-1) \end{bmatrix} \end{aligned}$$

4 RLS with $\boldsymbol{\lambda(t)}$

5 GERLS

5.1 Filtering based recursive generalized LS algorithm

Consider the following ARARX model

$$A(z^{-1})y(t) = B(z^{-1})u(t) + \frac{1}{C(z^{-1})}v(t) \tag{5.1}$$

Define the filtered input $u_f(t) := C(z^{-1})u(t)$ and filtered output $y_f(t) = C(z^{-1})y(t)$. Multiplying the both sides of (5.1) by $C(z^{-1})$, yields

$$A(z^{-1})y_f(t) = B(z^{-1})u_f(t) + v(t)$$

For the above equation error type model, one has

$$y_f(t) = \varphi_f^T(t)\theta_s + v(t) \tag{5.2}$$

where

$$\begin{aligned} \varphi_f(t) &= [\ -y_f(t-1) \quad -y_f(t-2) \quad \dots \quad -y_f(t-n_a) \quad u_f(t-1) \quad u_f(t-2) \quad \dots \quad u_f(t-n_b)\ ]^T \\ \theta_s &= [\ a_1 \quad a_2 \quad \dots \quad a_{n_a} \quad b_1 \quad b_2 \quad \dots \quad b_{n_b}\ ]^T \end{aligned}$$

Next, we will construct another algorithm to estimate $C(z^{-1})$. Define an intermediate variable

$$w(t)=\frac{1}{C(z^{-1})}v(t) \tag{5.3}$$

which gives

$$C(z^{-1})w(t)=v(t)$$

By this relation, it can be obtained

$$w(t)=\varphi_n^T(t)\theta_n+v(t) \tag{5.4}$$

where

$$\begin{array}{rcl}\varphi_n(t) & = & [\ -\omega(t-1)\ \ -\omega(t-2)\ \ \dots\ \ -\omega(t-n_c)\ ]^T \\ \theta_n & = & [\ c_1\ \ c_2\ \ \dots\ \ c_{n_c}\ ]^T\end{array}$$

It follows from (5.1) and (5.3) that

$$\begin{array}{rcl}w(t) & = & A(z^{-1})y(t)-B(z^{-1})u(t) \\ & = & y(t)-\varphi_s^T\theta_s\end{array}$$

where $\varphi_s(t)\ =\ [\ -y(t-1)\ \ -y(t-2)\ \ \dots\ \ -y(t-n_a)\ \ u(t-1)\ \ u(t-2)\ \ \dots\ \ u(t-n_b)\ ]^T$.

Then we can obtain an estimate of $w(t)$ as

$$\hat{w}(t)=y(t)-\hat{\varphi}_s^T(t)\hat{\theta}_s(t-1)$$

According to (5.2) and (5.4), two least square algorithms can be construct

$$\left\{\begin{array}{rcl}\hat{\theta}_s(t) & = & \hat{\theta}_s(t-1)+L_s(t)\big[\,\hat{y}_f(t)-\hat{\varphi}_f^T(t)\hat{\theta}_s(t)\big] \\ L_s(t) & = & \frac{P_s(t-1)\hat{\varphi}_f(t)}{1+\hat{\varphi}_f^T(t)P_s(t-1)\hat{\varphi}_f(t)} \\ P_s(t) & = & [I-L_n(t)\hat{\varphi}_s^T(t)]P_s(t-1),\,P_s(0)=p_0I\end{array}\right. \tag{5.5}$$

$$\left\{\begin{array}{rcl}\hat{\theta}_n(t) & = & \hat{\theta}_n(t-1)+L_n(t)\big[\,\hat{w}(t)-\hat{\varphi}_n^T(t)\hat{\theta}_n(t)\big] \\ L_n(t) & = & \frac{P_n(t-1)\hat{\varphi}_n(t)}{1+\hat{\varphi}_n^T(t)P_n(t-1)\hat{\varphi}_n(t)} \\ P_n(t) & = & [I-L_n(t)\hat{\varphi}_n^T(t)]P_n(t-1),\,P_n(0)=p_0I\end{array}\right. \tag{5.6}$$

where

$$\begin{array}{rcl}\hat{\theta}_n & = & [\ \hat{c}_1(t)\ \ \hat{c}_2(t)\ \ \dots\ \ \hat{c}_{n_c}(t)\ ]^T \\ \hat{\theta}_s & = & \big[\ \hat{a}_1(t)\ \ \hat{a}_2(t)\ \ \dots\ \ \hat{a}_{n_a}(t)\ \ \hat{b}_1(t)\ \ \hat{b}_2(t)\ \ \dots\ \ \hat{b}_{n_b}(t)\ \big]^T \\ \hat{\varphi}_n(t) & = & [\ -\hat{\omega}(t-1)\ \ -\hat{\omega}(t-2)\ \ \dots\ \ -\hat{\omega}(t-n_c)\ ]^T \\ \varphi_s(t) & = & [\ -y(t-1)\ \ -y(t-2)\ \ \dots\ \ -y(t-n_a)\ \ u(t-1)\ \ u(t-2)\ \ \dots\ \ u(t-n_b)\ ]^T \\ \hat{\varphi}_f(t) & = & [\ -\hat{y}_f(t-1)\ \ -\hat{y}_f(t-2)\ \ \dots\ \ -\hat{y}_f(t-n_a)\ \ \hat{u}_f(t-1)\ \ \hat{u}_f(t-2)\ \ \dots\ \ \hat{u}_f(t-n_b)\ ]^T \\ \hat{u}_f(t) & = & u(t)+\hat{c}_1(t)u(t-1)+\cdots+\hat{c}_{n_c}(t)u(t-n_c) \\ \hat{y}_f(t) & = & y(t)+\hat{c}_1(t)y(t-1)+\cdots+\hat{c}_{n_c}(t)y(t-n_c) \\ \hat{w}(t) & = & y(t)-\hat{\varphi}_s^T(t)\hat{\theta}_s(t-1)\end{array}$$

5.2 Filtering based recursive generalized extended LS algorithm

Consider the following ARARX model

$$A(z^{-1})y(t)=B(z^{-1})u(t)+\frac{D(z^{-1})}{C(z^{-1})}v(t) \tag{5.7}$$

Define the filtered input $u_f(t){:=}\frac{D(z^{-1})}{C(z^{-1})}u(t)$ and filtered output $y_f(t)=\frac{D(z^{-1})}{C(z^{-1})}y(t)$. Multiplying the both sides of (5.7) by $\frac{C(z^{-1})}{D(z^{-1})}$, yields

$$A(z^{-1})y_f(t)=B(z^{-1})u_f(t)+v(t)$$

For the above equation error type model, one has

$$y_f(t)=\varphi_f^T(t)\theta_s+v(t) \tag{5.8}$$

where

$$\begin{aligned}\theta_s &= [\ a_1\ \ a_2\ \ \dots\ \ a_{n_a}\ \ b_1\ \ b_2\ \ \dots\ \ b_{n_b}\ ]^T\\ \varphi_f(t) &= [\ -y_f(t-1)\ \ -y_f(t-2)\ \ \dots\ \ -y_f(t-n_a)\ \ u_f(t-1)\ \ u_f(t-2)\ \ \dots\ \ u_f(t-n_b)\ ]^T\end{aligned}$$

Next, we will construct another algorithm to estimate $\frac{D(z^{-1})}{C(z^{-1})}$. Define an intermediate variable

$$w(t)=\frac{D(z^{-1})}{C(z^{-1})}v(t) \tag{5.9}$$

which gives

$$C(z^{-1})w(t)=D(z^{-1})v(t)$$

By this relation, it can be obtained

$$w(t)=\varphi_n^T(t)\theta_n+v(t) \tag{5.10}$$

where

$$\begin{aligned}\theta_n &= [\ c_1\ \ c_2\ \ \dots\ \ c_{n_c}\ \ d_1\ \ d_2\ \ \dots\ \ d_{n_d}\ ]^T\\ \varphi_n(t) &= [\ -\omega(t-1)\ \ -\omega(t-2)\ \ \dots\ \ -\omega(t-n_c)\ \ v(t-1)\ \ v(t-2)\ \ \dots\ \ v(t-n_d)\ ]^T\end{aligned}$$

It follows from (5.7) and (5.9) that

$$\begin{aligned}w(t) &= A(z^{-1})y(t)-B(z^{-1})u(t)\\ &= y(t)-\varphi_s^T\theta_s\end{aligned}$$

where $\varphi_s(t) = [\ -y(t-1)\ \ -y(t-2)\ \ \dots\ \ -y(t-n_a)\ \ u(t-1)\ \ u(t-2)\ \ \dots\ \ u(t-n_b)\ ]^T$.

Then we can obtain an estimate of $w(t)$ as

$$\hat{w}(t)=y(t)-\hat{\varphi}_s^T(t)\hat{\theta}_s(t-1)$$

According to (5.8) and (5.10), two least square algorithms can be construct

$$\begin{cases} \hat{\theta}_s(t) &= \hat{\theta}_s(t-1) + L_s(t)\big[\, \hat{y}_f(t) - \hat{\varphi}_f^T(t)\hat{\theta}_s(t) \,\big] \\ L_s(t) &= \frac{P_s(t-1)\hat{\varphi}_f(t)}{1+\hat{\varphi}_f^T(t)P_s(t-1)\hat{\varphi}_f(t)} \\ P_s(t) &= [I - L_n(t)\hat{\varphi}_s^T(t)]P_s(t-1), P_s(0) = p_0 I \end{cases} \tag{5.11}$$

$$\begin{cases} \hat{\theta}_n(t) &= \hat{\theta}_n(t-1) + L_n(t)\big[\, \hat{w}(t) - \hat{\varphi}_n^T(t)\hat{\theta}_n(t) \,\big] \\ L_n(t) &= \frac{P_n(t-1)\hat{\varphi}_n(t)}{1+\hat{\varphi}_n^T(t)P_n(t-1)\hat{\varphi}_n(t)} \\ P_n(t) &= [I - L_n(t)\hat{\varphi}_n^T(t)]P_n(t-1), P_n(0) = p_0 I \end{cases} \tag{5.12}$$

where

$$\begin{aligned} \hat{\theta}_s &= \left[\ \hat{a}_1(t) \ \ \hat{a}_2(t) \ \ \ldots \ \ \hat{a}_{n_a}(t) \ \ \hat{b}_1(t) \ \ \hat{b}_2(t) \ \ \ldots \ \ \hat{b}_{n_b}(t)\ \right]^T \\ \hat{\theta}_n &= \left[\ \hat{c}_1(t) \ \ \hat{c}_2(t) \ \ \ldots \ \ \hat{c}_{n_c}(t) \ \ \hat{d}_1(t) \ \ \hat{d}_2(t) \ \ \ldots \ \ \hat{d}_{n_d}(t)\ \right]^T \\ \varphi_s(t) &= [\, -y(t-1) \ \ -y(t-2) \ \ \ldots \ \ -y(t-n_a) \ \ u(t-1) \ \ u(t-2) \ \ \ldots \ \ u(t-n_b) \,]^T \\ \hat{\varphi}_n(t) &= [\, -\hat{\omega}(t-1) \ \ -\hat{\omega}(t-2) \ \ \ldots \ \ -\hat{\omega}(t-n_c) \ \ \hat{v}(t-1) \ \ \hat{v}(t-2) \ \ \ldots \ \ \hat{v}(t-n_d) \,]^T \\ \hat{\varphi}_f(t) &= [\, -\hat{y}_f(t-1) \ \ -\hat{y}_f(t-2) \ \ \ldots \ \ -\hat{y}_f(t-n_a) \ \ \hat{u}_f(t-1) \ \ \hat{u}_f(t-2) \ \ \ldots \ \ \hat{u}_f(t-n_b) \,]^T \\ \hat{u}_f(t) &= -\hat{d}_1(t)\hat{u}_f(t-1) - \cdots - \hat{d}_{n_d}(t)\hat{u}_f(t-n_d) + u(t) + \hat{c}_1(t)u(t-1) + \cdots + \hat{c}_{n_c}(t)u(t-n_c) \\ \hat{y}_f(t) &= -\hat{d}_1(t)\hat{y}_f(t-1) - \cdots - \hat{d}_{n_d}(t)\hat{y}_f(t-n_d) + y(t) + \hat{c}_1(t)y(t-1) + \cdots + \hat{c}_{n_c}(t)y(t-n_c) \\ \hat{w}(t) &= y(t) - \hat{\varphi}_s^T(t)\hat{\theta}_s(t-1) \end{aligned}$$