Data filtering based recursive least squares parameter estimation for ARMAX models

Yuwu Liao
Dept. of Physics and Electronics Information Technology,
Xiangfan University,
Xiangfan, China 441053
168.lyw@163.com.

Dongqing Wang
College of Automation Engineering,
Qingdao University,
Qingdao, China 266071
dqwang64@163.com

Feng Ding
School of Communication and Control Engineering,
Jiangnan University,
Wuxi, China 214122
fding@jiangnan.edu.cn

Abstract

This paper uses an estimated noise transfer function to filter the input-output data and presents a filtering based recursive least squares algorithm for ARMAX models. Through the data filtering, we obtain two identification models, one including the parameters of the system model, and the other including the parameters of the noise model. Thus, the recursive least squares method can estimate the parameters of these two identification models, respectively, by replacing unmeasurable noise terms in the information vectors with their estimates. The proposed F-RLS algorithm has high computational efficiency because the dimensions of its covariance matrices become small and can generate more accurate parameter estimation compared with other existing algorithms.

1 Introduction

Consider a CARMA model (Controlled Auto-Regressive Moving Average model), or called ARMAX model (Auto-Regressive Moving Average model with eXogenous input) [1], depicted in Figure 1,

$$A(z)y(t) = B(z)u(t) + D(z)v(t), \qquad (1)$$

where $u(t)$ and $y(t)$ are the system input and output, respectively, $v(t)$ is a stochastic white noise with zero mean and variance σ^2, the disturbance $e(t) := D(z)v(t)$ is an MA model, $A(z)$, $B(z)$ and $D(z)$ are polynomials in z^{-1}, and defined by

$$A(z) = 1 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_{n_a} z^{-n_a},$$

$$B(z) = b_1 z^{-1} + b_2 z^{-2} + \cdots + b_{n_b} z^{-n_b},$$
$$D(z) = 1 + d_1 z^{-1} + d_2 z^{-2} + \cdots + d_{n_d} z^{-n_d}.$$

Assume that the degrees n_a, n_b and n_d are known and $y(t) = 0$, $u(t) = 0$ and $v(t) = 0$ for $t \leq 0$.

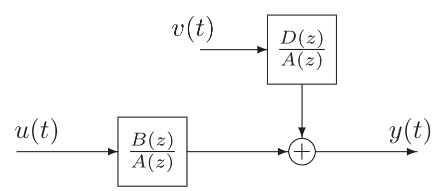

Figure 1. The ARMAX system

For special cases of the system in (1), many approaches can estimate their parameters. For example, when $D(z) = 1$, the system in (1) reduces to an equation error model, i.e., CAR model (Controlled Auto-Regressive model), or called ARX model (Auto-Regressive model with eXogenous input),

$$A(z)y(t) = B(z)u(t) + v(t),$$

for which the recursive least squares algorithm can estimate its parameters a_i and b_i [1–3]. The recursive extended least squares algorithm or prediction error methods can identify the parameters a_i, b_i and d_i of the ARMAX systems in (1) [1, 2, 4] and can obtain the parameter estimates of both system models and noise models.

Although the instrumental variable least squares and bias compensation/correction least squares algorithms can identify the systems in (1) [1, 5–10], the disadvantages are that they fail to obtain the parameter estimates of the noise models.

This paper discussed identification problems for ARMAX systems based on the input-output data filtering technique. The objective is to present a filtering based recursive least squares algorithm (F-RLS) to estimate the system

[0]This work was supported by Colleges and Universities In Shandong Province of Outstanding Young Teachers Visiting Scholar at the Domestic-Funded Projects.

978-0-7695-3501-2/09 $25.00 © 2009 IEEE
DOI 10.1109/CMC.2009.140

parameters (a_i, b_i, c_i, d_i) from available input-output data $\{u(t), y(t)\}$ and to evaluate the accuracy of the parameter estimates by simulations on computers.

Briefly, the paper is organized as follows. Section 2 simply gives the RELS algorithm for ARMAX systems. Section 3 derives a filtering based recursive least squares algorithm for ARMAX systems. Section 4 provides an illustrative example for the results in this paper. Finally, concluding remarks are given in Section 5.

2 The RELS algorithms

To show the advantages of the F-RLS algorithm we will propose, the following gives the recursive extended least squares algorithm for comparisons.

Define the parameter vector $\boldsymbol{\theta}$ and the information vector $\boldsymbol{\varphi}_0(t)$ as

$$\begin{aligned}
\boldsymbol{\theta} &:= \begin{bmatrix} \boldsymbol{\theta}_{\mathrm{s}} \\ \boldsymbol{\theta}_{\mathrm{n}} \end{bmatrix} \in \mathbb{R}^n, \\
\boldsymbol{\theta}_{\mathrm{s}} &:= [a_1, a_2, \cdots, a_{n_a}, b_1, b_2, \cdots, b_{n_b}]^{\mathrm{T}} \in \mathbb{R}^{n_a+n_b}, \\
\boldsymbol{\theta}_{\mathrm{n}} &:= [d_1, d_2, \cdots, d_{n_d}]^{\mathrm{T}} \in \mathbb{R}^{n_d}, \\
\boldsymbol{\varphi}_0(t) &= \begin{bmatrix} \boldsymbol{\varphi}_{\mathrm{s}}(t) \\ \boldsymbol{\varphi}_{\mathrm{n}}(t) \end{bmatrix} \in \mathbb{R}^n, \\
\boldsymbol{\varphi}_{\mathrm{s}}(t) &:= [-y(t-1), -y(t-2), \cdots, -y(t-n_a), \\
&\qquad u(t-1), u(t-2), \cdots, u(t-n_b)]^{\mathrm{T}} \in \mathbb{R}^{n_a+n_d}, \\
\boldsymbol{\varphi}_{\mathrm{n}}(t) &:= [v(t-1), v(t-2), \cdots, v(t-n_d)]^{\mathrm{T}} \in \mathbb{R}^{n_d},
\end{aligned}$$

Thus, we have

$$\begin{aligned}
e(t) &= D(z)v(t) \\
&= \boldsymbol{\varphi}_{\mathrm{n}}^{\mathrm{T}}(t)\boldsymbol{\theta}_{\mathrm{n}} + v(t), \qquad (2) \\
y(t) &= [1 - A(z)]y(t) + B(z)u(t) + e(t) \\
&= \boldsymbol{\varphi}_{\mathrm{s}}^{\mathrm{T}}(t)\boldsymbol{\theta}_{\mathrm{s}} + e(t) \\
&= \boldsymbol{\varphi}_{\mathrm{s}}^{\mathrm{T}}(t)\boldsymbol{\theta}_{\mathrm{s}} + \boldsymbol{\varphi}_{\mathrm{n}}^{\mathrm{T}}(t)\boldsymbol{\theta}_{\mathrm{n}} + v(t) \\
&=: \boldsymbol{\varphi}_0^{\mathrm{T}}(t)\boldsymbol{\theta} + v(t), \qquad (3)
\end{aligned}$$

Because the information vector $\boldsymbol{\varphi}_{\mathrm{n}}(t)$ in $\boldsymbol{\varphi}_0(t)$ on the right-hand sides contains unmeasurable noise terms $v(t-i)$, the following standard recursive least squares algorithm cannot generate the estimate of the parameter vector $\boldsymbol{\theta}$,

$$\hat{\boldsymbol{\theta}}(t) = \hat{\boldsymbol{\theta}}(t-1) + \boldsymbol{L}(t)[y(t) - \boldsymbol{\varphi}_0^{\mathrm{T}}(t)\hat{\boldsymbol{\theta}}(t-1)], \qquad (4)$$

$$\boldsymbol{L}(t) = \frac{\boldsymbol{P}(t-1)\boldsymbol{\varphi}_0(t)}{1 + \boldsymbol{\varphi}_0^{\mathrm{T}}(t)\boldsymbol{P}(t-1)\boldsymbol{\varphi}_0(t)}, \qquad (5)$$

$$\boldsymbol{P}(t) = [\boldsymbol{I} - \boldsymbol{L}(t)\boldsymbol{\varphi}_0^{\mathrm{T}}(t)]\boldsymbol{P}(t-1),\ \boldsymbol{P}(0) = p_0\boldsymbol{I}. \qquad (6)$$

The solution is to replace these unmeasurable noise terms $v(t-i)$ in $\boldsymbol{\varphi}_{\mathrm{n}}(t)$ of $\boldsymbol{\varphi}(t)$ with their estimated residuals $\hat{v}(t-i)$ and define

$$\begin{aligned}
\hat{\boldsymbol{\varphi}}_{\mathrm{n}}(t) &:= [\hat{v}(t-1), \hat{v}(t-2), \cdots, \hat{v}(t-n_d)]^{\mathrm{T}} \in \mathbb{R}^{n_d}, \\
\boldsymbol{\varphi}(t) &:= \begin{bmatrix} \boldsymbol{\varphi}_{\mathrm{s}}(t) \\ \hat{\boldsymbol{\varphi}}_{\mathrm{n}}(t) \end{bmatrix}.
\end{aligned}$$

Let $\hat{\boldsymbol{\theta}}(t) = \begin{bmatrix} \hat{\boldsymbol{\theta}}_{\mathrm{s}}(t) \\ \hat{\boldsymbol{\theta}}_{\mathrm{n}}(t) \end{bmatrix}$ be the estimate of $\boldsymbol{\theta} = \begin{bmatrix} \boldsymbol{\theta}_{\mathrm{s}} \\ \boldsymbol{\theta}_{\mathrm{n}} \end{bmatrix}$. Replacing $\boldsymbol{\varphi}_0(t)$ and $\boldsymbol{\theta}$ in (3) with $\boldsymbol{\varphi}(t)$ and $\hat{\boldsymbol{\theta}}(t)$, respectively, the estimate $\hat{v}(t)$ can be computed by

$$\hat{v}(t) = y(t) - \boldsymbol{\varphi}^{\mathrm{T}}(t)\hat{\boldsymbol{\theta}}(t).$$

Note that $\boldsymbol{\varphi}(t)$ is known at time t. Replacing $\boldsymbol{\varphi}_0(t)$ in (4)-(6) with $\boldsymbol{\varphi}(t)$ yields a recursive extended least squares algorithm (RELS) to identify the parameters of the ARMAX model in (3):

$$\hat{\boldsymbol{\theta}}(t) = \hat{\boldsymbol{\theta}}(t-1) + \boldsymbol{L}(t)[y(t) - \boldsymbol{\varphi}^{\mathrm{T}}(t)\hat{\boldsymbol{\theta}}(t-1)], \qquad (7)$$

$$\boldsymbol{L}(t) = \frac{\boldsymbol{P}(t-1)\boldsymbol{\varphi}(t)}{1 + \boldsymbol{\varphi}^{\mathrm{T}}(t)\boldsymbol{P}(t-1)\boldsymbol{\varphi}(t)}, \qquad (8)$$

$$\boldsymbol{P}(t) = [\boldsymbol{I} - \boldsymbol{L}(t)\boldsymbol{\varphi}^{\mathrm{T}}(t)]\boldsymbol{P}(t-1), \qquad (9)$$

$$\hat{\boldsymbol{\theta}}(t) = \begin{bmatrix} \hat{\boldsymbol{\theta}}_{\mathrm{s}}(t) \\ \hat{\boldsymbol{\theta}}_{\mathrm{n}}(t) \end{bmatrix}, \quad \boldsymbol{\varphi}(t) = \begin{bmatrix} \boldsymbol{\varphi}_{\mathrm{s}}(t) \\ \hat{\boldsymbol{\varphi}}_{\mathrm{n}}(t) \end{bmatrix}, \qquad (10)$$

$$\begin{aligned}
\boldsymbol{\varphi}_{\mathrm{s}}(t) &= [-y(t-1), -y(t-2), \cdots, -y(t-n_a), \\
&\qquad u(t-1), u(t-2), \cdots, u(t-n_b)]^{\mathrm{T}}, \qquad (11)
\end{aligned}$$

$$\hat{\boldsymbol{\varphi}}_{\mathrm{n}}(t) = [\hat{v}(t-1), \hat{v}(t-2), \cdots, \hat{v}(t-n_d)]^{\mathrm{T}}, \qquad (12)$$

$$\hat{v}(t) = y(t) - \boldsymbol{\varphi}^{\mathrm{T}}(t)\hat{\boldsymbol{\theta}}(t). \qquad (13)$$

3 The filtering based recursive least squares algorithm

If the input-output data are filtered through the rational fraction $\frac{1}{D(z)}$ (a linear filter), model (1) becomes "an equation error model", then the recursive least squares algorithm can be applied. Because $\frac{1}{D(z)}$ is unknown, its estimate $\frac{1}{\hat{D}(t,z)}$ is generally used to filter the input-output data. The identification method based on this idea is called the filtering based recursive least squares one (F-RLS).

For the ARMAX system in (1), define the filtered input $u_{\mathrm{f}}(t)$, filtered output $y_{\mathrm{f}}(t)$ and filtered information vector $\boldsymbol{\varphi}_{\mathrm{f}}(t)$ as

$$u_{\mathrm{f}}(t) := \frac{1}{D(z)}\, u(t), \quad y_{\mathrm{f}}(t) := \frac{1}{D(z)}\, y(t), \qquad (14)$$

$$\begin{aligned}
\boldsymbol{\varphi}_{\mathrm{f}}(t) &:= [-y_{\mathrm{f}}(t-1), \cdots, -y_{\mathrm{f}}(t-n_a), \\
&\qquad u_{\mathrm{f}}(t-1), \cdots, u_{\mathrm{f}}(t-n_b)]^{\mathrm{T}} \in \mathbb{R}^{n_a+n_b}. \qquad (15)
\end{aligned}$$

Dividing both sides of (1) by $D(z)$ gives

$$A(z)\frac{1}{D(z)}\, y(t) = B(z)\frac{1}{D(z)}\, u(t) + v(t),$$

or

$$A(z)y_{\mathrm{f}}(t) = B(z)u_{\mathrm{f}}(t) + v(t).$$

This filtered model is an equation error model (CAR/ARX model) and can be rewritten in a vector form,

$$\begin{aligned}
y_{\mathrm{f}}(t) &= [1 - A(z)]y_{\mathrm{f}}(t) + B(z)u_{\mathrm{f}}(t) + v(t) \\
&= \boldsymbol{\varphi}_{\mathrm{f}}^{\mathrm{T}}(t)\boldsymbol{\theta}_{\mathrm{s}} + v(t). \qquad (16)
\end{aligned}$$

Define the inner variable:

$$e(t) := D(z)v(t). \tag{17}$$

or

$$e(t) = \boldsymbol{\varphi}_{\mathrm{n}}^{\mathrm{T}}(t)\boldsymbol{\theta}_{\mathrm{n}} + v(t). \tag{18}$$

For two identification models (16) and (18), using the following two least squares algorithms cannot generate the estimates $\hat{\boldsymbol{\theta}}_{\mathrm{s}}(t)$ and $\hat{\boldsymbol{\theta}}_{\mathrm{n}}(t)$ of $\boldsymbol{\theta}$,

$$\hat{\boldsymbol{\theta}}_{\mathrm{s}}(t) = \hat{\boldsymbol{\theta}}_{\mathrm{s}}(t-1) + \boldsymbol{L}_{\mathrm{f}}(t)[y_{\mathrm{f}}(t) - \boldsymbol{\varphi}_{\mathrm{f}}^{\mathrm{T}}(t)\hat{\boldsymbol{\theta}}_{\mathrm{s}}(t-1)], \tag{19}$$

$$\boldsymbol{L}_{\mathrm{f}}(t) = \frac{\boldsymbol{P}_{\mathrm{f}}(t-1)\boldsymbol{\varphi}_{\mathrm{f}}(t)}{1 + \boldsymbol{\varphi}_{\mathrm{f}}^{\mathrm{T}}(t)\boldsymbol{P}_{\mathrm{f}}(t-1)\boldsymbol{\varphi}_{\mathrm{f}}(t)}, \tag{20}$$

$$\boldsymbol{P}_{\mathrm{f}}(t) = [\boldsymbol{I} - \boldsymbol{L}_{\mathrm{f}}(t)\boldsymbol{\varphi}_{\mathrm{f}}^{\mathrm{T}}(t)]\boldsymbol{P}_{\mathrm{f}}(t-1), \tag{21}$$

$$\hat{\boldsymbol{\theta}}_{\mathrm{n}}(t) = \hat{\boldsymbol{\theta}}_{\mathrm{n}}(t-1) + \boldsymbol{L}_{\mathrm{n}}(t)[e(t) - \boldsymbol{\varphi}_{\mathrm{n}}^{\mathrm{T}}(t)\hat{\boldsymbol{\theta}}_{\mathrm{n}}(t-1)], \tag{22}$$

$$\boldsymbol{L}_{\mathrm{n}}(t) = \frac{\boldsymbol{P}_{\mathrm{n}}(t-1)\boldsymbol{\varphi}_{\mathrm{n}}(t)}{1 + \boldsymbol{\varphi}_{\mathrm{n}}^{\mathrm{T}}(t)\boldsymbol{P}_{\mathrm{n}}(t-1)\boldsymbol{\varphi}_{\mathrm{n}}(t)}, \tag{23}$$

$$\boldsymbol{P}_{\mathrm{n}}(t) = [\boldsymbol{I} - \boldsymbol{L}_{\mathrm{n}}(t)\boldsymbol{\varphi}_{\mathrm{n}}^{\mathrm{T}}(t)]\boldsymbol{P}_{\mathrm{n}}(t-1). \tag{24}$$

Because the polynomial $D(z)$ is unknown, then $u_{\mathrm{f}}(t)$ and $y_{\mathrm{f}}(t)$ are unknown, the information vector $\boldsymbol{\varphi}_{\mathrm{f}}(t)$ and $\boldsymbol{\varphi}_{\mathrm{n}}(t)$ are unknown, the algorithms in (19)-(24) are impossible to realize. Here, we still adopt the idea of replacing the unknown variables with their estimates to derive our F-RLS identification algorithms.

Since

$$\begin{aligned} e(t) &= A(z)y(t) - B(z)u(t) \\ &= y(t) - \boldsymbol{\varphi}_{\mathrm{s}}^{\mathrm{T}}(t)\boldsymbol{\theta}_{\mathrm{s}}. \end{aligned} \tag{25}$$

From the above equation and (18), we get

$$\begin{aligned} y(t) &= \boldsymbol{\varphi}_{\mathrm{s}}^{\mathrm{T}}(t)\boldsymbol{\theta}_{\mathrm{s}} + e(t) \\ &=: \boldsymbol{\varphi}_{0}^{\mathrm{T}}(t)\boldsymbol{\theta} + v(t) \end{aligned} \tag{26}$$

Replacing the unknown $\boldsymbol{\theta}_{\mathrm{s}}$ on the right-hand side of (25) with the estimate $\hat{\boldsymbol{\theta}}_{\mathrm{s}}(t)$, the estimate $\hat{e}(t)$ can be computed by

$$\hat{e}(t) = y(t) - \boldsymbol{\varphi}_{\mathrm{s}}^{\mathrm{T}}(t)\hat{\boldsymbol{\theta}}_{\mathrm{s}}(t).$$

Let the estimate of $v(t)$ be $\hat{v}(t)$ and use $\hat{v}(t-i)$ to construct the estimate of $\boldsymbol{\varphi}_{\mathrm{n}}(t)$ as follows:

$$\hat{\boldsymbol{\varphi}}_{\mathrm{n}}(t) = [\hat{v}(t-1), \hat{v}(t-2), \cdots, \hat{v}(t-n_d)]^{\mathrm{T}} \in \mathbb{R}^{n_d}.$$

From (18), we have

$$v(t) = e(t) - \boldsymbol{\varphi}_{\mathrm{n}}^{\mathrm{T}}(t)\boldsymbol{\theta}_{\mathrm{n}}.$$

Replacing $\boldsymbol{\varphi}_{\mathrm{n}}(t)$ and $\boldsymbol{\theta}_{\mathrm{n}}$ in the above equation with $\hat{\boldsymbol{\varphi}}_{\mathrm{n}}(t)$ and $\hat{\boldsymbol{\theta}}_{\mathrm{n}}(t)$, the estimate $\hat{e}(t)$ can be computed by

$$\hat{v}(t) = \hat{e}(t) - \hat{\boldsymbol{\varphi}}_{\mathrm{n}}^{\mathrm{T}}(t)\hat{\boldsymbol{\theta}}_{\mathrm{n}}(t).$$

Using the parameter estimates of the noise model,

$$\hat{\boldsymbol{\theta}}_{\mathrm{n}}(t) = [\hat{d}_1(t), \hat{d}_2(t), \cdots, \hat{d}_{n_d}(t)]^{\mathrm{T}}$$

to construct the estimates of $C(z)$ and $D(z)$,

$$\hat{D}(t,z) = 1 + \hat{d}_1(t)z^{-1} + \hat{d}_2(t)z^{-2} + \cdots + \hat{d}_{n_d}(t)z^{-n_d}.$$

Filtering $u(t)$ and $y(t)$ with $\frac{1}{\hat{D}(t,z)}$ to get the estimates of $u_{\mathrm{f}}(t)$ and $y_{\mathrm{f}}(t)$ as follows:

$$\hat{u}_{\mathrm{f}}(t) = \frac{1}{\hat{D}(t,z)}u(t), \quad \hat{y}_{\mathrm{f}}(t) = \frac{1}{\hat{D}(t,z)}y(t).$$

or

$$\begin{aligned} \hat{D}(t,z)\hat{u}_{\mathrm{f}}(t) &= u(t), \\ \hat{D}(t,z)\hat{y}_{\mathrm{f}}(t) &= y(t). \end{aligned}$$

Also $\hat{u}_{\mathrm{f}}(t)$ and $\hat{y}_{\mathrm{f}}(t)$ can be recursively computed by

$$\begin{aligned} \hat{u}_{\mathrm{f}}(t) &= [1 - \hat{D}(t,z)]\hat{u}_{\mathrm{f}}(t) + u(t) \\ &= -\hat{d}_1(t)\hat{u}_{\mathrm{f}}(t-1) - \cdots - \hat{d}_{n_d}(t)\hat{u}_{\mathrm{f}}(t-n_d) + u(t), \\ \hat{y}_{\mathrm{f}}(t) &= [1 - \hat{D}(t,z)]\hat{y}_{\mathrm{f}}(t) + y(t) \\ &= -\hat{d}_1(t)\hat{y}_{\mathrm{f}}(t-1) - \cdots - \hat{d}_{n_d}(t)\hat{y}_{\mathrm{f}}(t-n_d) + y(t). \end{aligned}$$

Construct the estimate of $\boldsymbol{\varphi}_{\mathrm{f}}(t)$ with $\hat{y}_{\mathrm{f}}(t-i)$ and $\hat{u}_{\mathrm{f}}(t-i)$ as follows:

$$\begin{aligned} \hat{\boldsymbol{\varphi}}_{\mathrm{f}}(t) = [&-\hat{y}_{\mathrm{f}}(t-1), -\hat{y}_{\mathrm{f}}(t-2), \cdots, -\hat{y}_{\mathrm{f}}(t-n_a), \\ &\hat{u}_{\mathrm{f}}(t-1), \hat{u}_{\mathrm{f}}(t-2), \cdots, \hat{u}_{\mathrm{f}}(t-n_b)]^{\mathrm{T}} \in \mathbb{R}^{n_a+n_b}, \end{aligned}$$

Replacing the unknown information vector $\boldsymbol{\varphi}_{\mathrm{f}}(t)$ in (19)-(21) with $\hat{\boldsymbol{\varphi}}_{\mathrm{f}}(t)$, $y_{\mathrm{f}}(t)$ in (19) with $\hat{y}_{\mathrm{f}}(t)$, $\boldsymbol{\varphi}_{\mathrm{n}}(t)$ in (22)-(24) with $\hat{\boldsymbol{\varphi}}_{\mathrm{n}}(t)$, and the unknown noise terms $e(t)$ in (22) with $\hat{e}(t)$, we obtain the filtering based recursive least squares algorithms (F-RLS) of estimating the parameter vectors $\boldsymbol{\theta}_{\mathrm{s}}$ and $\boldsymbol{\theta}_{\mathrm{n}}$ for the ARMAX systems:

$$\hat{\boldsymbol{\theta}}_{\mathrm{s}}(t) = \hat{\boldsymbol{\theta}}_{\mathrm{s}}(t-1) + \boldsymbol{L}_{\mathrm{f}}(t)[\hat{y}_{\mathrm{f}}(t) - \hat{\boldsymbol{\varphi}}_{\mathrm{f}}^{\mathrm{T}}(t)\hat{\boldsymbol{\theta}}_{\mathrm{s}}(t-1)], \tag{27}$$

$$\boldsymbol{L}_{\mathrm{f}}(t) = \frac{\boldsymbol{P}_{\mathrm{f}}(t-1)\hat{\boldsymbol{\varphi}}_{\mathrm{f}}(t)}{1 + \hat{\boldsymbol{\varphi}}_{\mathrm{f}}^{\mathrm{T}}(t)\boldsymbol{P}_{\mathrm{f}}(t-1)\hat{\boldsymbol{\varphi}}_{\mathrm{f}}(t)}, \tag{28}$$

$$\boldsymbol{P}_{\mathrm{f}}(t) = [\boldsymbol{I} - \boldsymbol{L}_{\mathrm{f}}(t)\hat{\boldsymbol{\varphi}}_{\mathrm{f}}^{\mathrm{T}}(t)]\boldsymbol{P}_{\mathrm{f}}(t-1),\ \boldsymbol{P}_{\mathrm{f}}(0) = p_0\boldsymbol{I}, \tag{29}$$

$$\begin{aligned} \hat{\boldsymbol{\varphi}}_{\mathrm{f}}(t) = [&-\hat{y}_{\mathrm{f}}(t-1), -\hat{y}_{\mathrm{f}}(t-2), \cdots, -\hat{y}_{\mathrm{f}}(t-n_a), \\ &\hat{u}_{\mathrm{f}}(t-1), \hat{u}_{\mathrm{f}}(t-2), \cdots, \hat{u}_{\mathrm{f}}(t-n_b)]^{\mathrm{T}}, \end{aligned} \tag{30}$$

$$\hat{y}_{\mathrm{f}}(t) = -\hat{d}_1(t)\hat{y}_{\mathrm{f}}(t-1) - \cdots - \hat{d}_{n_d}(t)\hat{y}_{\mathrm{f}}(t-n_d) + y(t), \tag{31}$$

$$\hat{u}_{\mathrm{f}}(t) = -\hat{d}_1(t)\hat{y}_{\mathrm{f}}(t-1) - \cdots - \hat{d}_{n_d}(t)\hat{y}_{\mathrm{f}}(t-n_d) + u(t), \tag{32}$$

$$\hat{\boldsymbol{\theta}}_{\mathrm{n}}(t) = \hat{\boldsymbol{\theta}}_{\mathrm{n}}(t-1) + \boldsymbol{L}_{\mathrm{n}}(t)[\hat{e}(t) - \hat{\boldsymbol{\varphi}}_{\mathrm{n}}^{\mathrm{T}}(t)\hat{\boldsymbol{\theta}}_{\mathrm{n}}(t-1)], \tag{33}$$

$$\boldsymbol{L}_{\mathrm{n}}(t) = \frac{\boldsymbol{P}_{\mathrm{n}}(t-1)\hat{\boldsymbol{\varphi}}_{\mathrm{n}}(t)}{1 + \hat{\boldsymbol{\varphi}}_{\mathrm{n}}^{\mathrm{T}}(t)\boldsymbol{P}_{\mathrm{n}}(t-1)\hat{\boldsymbol{\varphi}}_{\mathrm{n}}(t)}, \tag{34}$$

$$\boldsymbol{P}_{\rm n}(t) = [\boldsymbol{I} - \boldsymbol{L}_{\rm n}(t)\hat{\boldsymbol{\varphi}}_{\rm n}^{\rm T}(t)]\boldsymbol{P}_{\rm n}(t-1),\ \boldsymbol{P}_{\rm n}(0) = p_0\boldsymbol{I}, \quad (35)$$

$$\hat{\boldsymbol{\varphi}}_{\rm n}(t) = [\hat{v}(t-1), \hat{v}(t-2), \cdots, \hat{v}(t-n_d)]^{\rm T}, \quad (36)$$

$$\hat{e}(t) = y(t) - \boldsymbol{\varphi}_{\rm s}^{\rm T}(t)\hat{\boldsymbol{\theta}}_{\rm s}(t), \quad (37)$$

$$\hat{v}(t) = \hat{e}(t) - \hat{\boldsymbol{\varphi}}_{\rm n}^{\rm T}(t)\hat{\boldsymbol{\theta}}_{\rm n}(t), \quad (38)$$

$$\boldsymbol{\varphi}_{\rm s}(t) = [-y(t-1), -y(t-2), \cdots, -y(t-n_a), u(t-1), u(t-2), \cdots, u(t-n_b)]^{\rm T}, \quad (39)$$

$$\hat{\boldsymbol{\theta}}_{\rm s}(t) = [\hat{a}_1(t), \cdots, \hat{a}_{n_a}(t),\ \hat{b}_1(t), \cdots, \hat{b}_{n_b}(t)]^{\rm T}, \quad (40)$$

$$\hat{\boldsymbol{\theta}}_{\rm n}(t) = [\hat{d}_1(t), \cdots, \hat{d}_{n_d}(t)]^{\rm T}. \quad (41)$$

The proposed F-RLS algorithm has high computational efficiency because the dimensions of its covariance matrices become small and can generate more accurate parameter estimation compared with the RELS algorithm.

To initialize the F-RLS algorithm, we take

$$\hat{\boldsymbol{\theta}}_{\rm s}(i) = \mathbf{1}_{n_a+n_b}/p_0,\ \hat{\boldsymbol{\theta}}_{\rm n}(i) = \mathbf{1}_{n_d}/p_0,\ i \leq 0, \quad (42)$$

$$\boldsymbol{P}_{\rm f}(0) = p_0\boldsymbol{I}_{n_a+n_b},\ \boldsymbol{P}_{\rm n}(0) = p_0\boldsymbol{I}_{n_d},\ p_0 = 10^6. \quad (43)$$

The steps involved in the F-RLS algorithms are list as follows:

1. Set $u(t) = 0$, $y(t) = 0$ for $t \leq 0$.
2. Let $t = 1$, set initial values of parameter estimation vectors and covariance matrices according to (42)-(43).
3. Compute $\hat{y}_{\rm f}(i)$ by (31) and $\hat{u}_{\rm f}(i)$ by (32), construct $\boldsymbol{\varphi}_{\rm s}(i)$ by (39), compute $\hat{e}(i) = 0$ for $i \leq 0$ by (37) and $\hat{v}(i)$ by (38).
4. Collect the input-output data $\{u(t), y(t)\}$, construct information vectors $\boldsymbol{\varphi}_{\rm s}(t)$ by (39), $\hat{\boldsymbol{\varphi}}_{\rm f}(t)$ by (30) and $\hat{\boldsymbol{\varphi}}_{\rm n}(t)$ by (36).
5. Compute the gain vector $\boldsymbol{L}_{\rm f}(t)$ by (28) and the covariance matrix $\boldsymbol{P}_{\rm f}(t)$ by (29).
6. Update the parameter estimate $\hat{\boldsymbol{\theta}}_{\rm s}(t)$ by (27).
7. Compute $\hat{e}(t)$ by (37), $\hat{v}(t)$ by (38), $\hat{y}_{\rm f}(t)$ by (31) and $\hat{u}_{\rm f}(t)$ by (32).
8. Compute the gain vector $\boldsymbol{L}_{\rm n}(t)$ by (34), the covariance matrix $\boldsymbol{P}_{\rm n}(t)$ by (35).
9. Update the parameter estimate $\hat{\boldsymbol{\theta}}_{\rm n}(t)$ by (33).
10. Compare $\hat{\boldsymbol{\theta}}_{\rm s}(t)$ with $\hat{\boldsymbol{\theta}}_{\rm s}(t-1)$, and $\hat{\boldsymbol{\theta}}_{\rm n}(t)$ with $\hat{\boldsymbol{\theta}}_{\rm n}(t-1)$, if they are sufficiently close, or for pre-set small positive constant $\epsilon > 0$, if

    $$\|\hat{\boldsymbol{\theta}}_{\rm s}(t) - \hat{\boldsymbol{\theta}}_{\rm s}(t-1)\| < \epsilon \text{ and } \|\hat{\boldsymbol{\theta}}_{\rm n}(t) - \hat{\boldsymbol{\theta}}_{\rm n}(t-1)\| < \epsilon,$$

    terminate the procedure, and obtain the parameter estimates $\hat{\boldsymbol{\theta}}_{\rm s}(t)$ and $\hat{\boldsymbol{\theta}}_{\rm n}(t)$; otherwise increment t by 1, go to step 4, and continue recursive computing.

4 Example

Consider the following stochastic system,

$$A(z)y(t) = B(z)u(t) + D(z)v(t),$$
$$A(z) = 1 + a_1z^{-1} + a_2z^{-2} = 1 + 1.50z^{-1} + 0.80z^{-2},$$
$$B(z) = b_1z^{-1} + b_2z^{-2} = 0.22z^{-1} + 1.80z^{-2},$$
$$D(z) = 1 + d_1z^{-1} = 1 - 0.10z^{-1},$$
$$\boldsymbol{\theta} = [a_1, a_2, b_1, b_2, c_1, d_1]^{\rm T} = [1.50, 0.80, 0.22, 1.80, -0.10]^{\rm T}.$$

The input $\{u(t)\}$ is taken as an uncorrelated persistent excitation signal sequence with zero mean and unit variance, and $\{v(t)\}$ as a white noise sequence with zero mean and variance $\sigma^2 = 0.50^2$. Applying the RELS and the F-RLS algorithms to estimate the parameters of this system, the parameter estimates and their errors are shown in Table 1 and the estimation errors $\delta := \|\hat{\boldsymbol{\theta}}(t) - \boldsymbol{\theta}\|/\|\boldsymbol{\theta}\|$ versus t are shown in Figure 2.

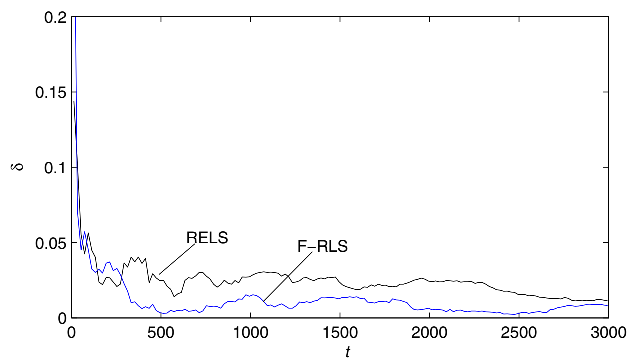

Figure 2. The estimation errors δ vs. t

From Table 1 and Figure 2, we can get the following conclusions:

- The parameter estimates given by the F-RLG algorithm converge to their true values fast compared with the RELS algorithm.
- The parameter estimation errors become (generally) smaller and smaller with the data length t increasing. This shows that the proposed algorithm is effective.

5 Conclusions

A filtering based recursive least squares algorithm for an ARMAX systems is derived by filtering the input-output data with an estimated transfer function. The proposed algorithms can require less computation and give highly accurate parameter estimates compared with the recursive extended least squares algorithms.

Table 1. The parameter estimates and their errors ($\sigma^2 = 0.50^2$)

Algorithms	t	a_1	a_2	b_1	b_2	d_1	δ (%)
RELS	100	1.51608	0.82413	0.17231	1.83460	-0.18736	4.39338
	200	1.52716	0.83508	0.18245	1.82662	-0.05085	3.24145
	500	1.51498	0.81975	0.20168	1.83135	-0.05701	2.47189
	1000	1.51090	0.81554	0.21259	1.79658	-0.03484	2.74807
	2000	1.51027	0.81276	0.20942	1.80989	-0.04230	2.48041
	3000	1.50858	0.80882	0.20943	1.80214	-0.07690	1.13785
F-RLS	100	1.44348	0.73554	0.21174	1.83492	-0.03567	4.54344
	200	1.49946	0.78603	0.27288	1.80789	-0.17941	3.88895
	500	1.49680	0.79624	0.22583	1.80115	-0.10023	0.31056
	1000	1.49692	0.79841	0.23022	1.79488	-0.06414	1.51923
	2000	1.49543	0.79487	0.22716	1.79132	-0.09050	0.65333
	3000	1.49676	0.79804	0.22243	1.80156	-0.12038	0.84121
True values		1.50000	0.80000	0.22000	1.80000	-0.10000	

References

[1] L. Ljung, "System Identification: Theory for the User (2nd edn)," Prentice-Hall: Englewood Cliffs, NJ, 1999.

[2] G.C. Goodwin and K.S. Sin, "Adaptive filtering prediction and control," Prentice-Hall: Englewood Cliffs, NJ, 1984.

[3] Y.S. Xiao, F. Ding, Y. Zhou, M. Li, and J.Y. Dai, "On consistency of recursive least squares identification algorithms for controlled auto-regression models," Applied Mathematical Modelling, vol. 32, no. 11, pp. 2207-2215, 2008.

[4] J. Ding and F. Ding, "The residual based extended least squares identification method for dual-rate systems," Computers & Mathematics with Applications, vol. 56, no. 6, pp. 1479-1487, 2008.

[5] W.X. Zheng, "A bias-correction method for indirect identification of closed-loop systems," Automatica, vol. 31, no. 7, pp. 1019-1024, 1995.

[6] W.X. Zheng, "Least-squares identification of a class of multivariable systems with correlated disturbances," Journal of the Franklin Institute, vol. 336, no. 8, pp. 1309-1324, 1999.

[7] W.X. Zheng, "A bias correction method for identification of linear dynamic errors-in-variables models," IEEE Transactions on Automatic Control, vol. 47, no. 7, pp. 1142-1147, 2002.

[8] W.X. Zheng, "On Indirect Identification of Feedback-Control Systems via the Instrumental Variables Methods," IEEE Transactions on Circuits and Systems-I: Fundamental Theory and Applications, vol. 50, no. 9, pp. 1232-1238, 2003.

[9] H.Z. Yang and Y. Zhang, "Comparisons of bias compensation methods and other identification approaches for Box-Jenkins models," Control Theory and Applications, vol. 24, no. 2, pp. 215-222, 2007.

[10] Y. Zhang and H.Z. Yang, "Bias compensation recursive least squares identification for output error systems with colored noises," Acta Automatica Sinica, vol. 33, no. 10, pp. 1053-1060, 2007.