Digital Signal Processing 20 (2010) 991–999

Contents lists available at ScienceDirect

Digital Signal Processing

www.elsevier.com/locate/dsp

Input–output data filtering based recursive least squares identification for CARARMA systems ☆

Dongqing Wang [a,*], Feng Ding [b]

[a] College of Automation Engineering, Qingdao University, Qingdao 266071, PR China
[b] School of Communication and Control Engineering, Jiangnan University, Wuxi 214122, PR China

ARTICLE INFO

Article history:
Available online 4 January 2010

Keywords:
Signal processing
Filtering
Recursive identification
Parameter estimation
Least squares

ABSTRACT

This paper uses an estimated noise transfer function to filter the input–output data and presents filtering based recursive least squares algorithms (F-RLS) for controlled autoregressive autoregressive moving average (CARARMA) systems. 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 be used to estimate the parameters of these two identification models, respectively, by replacing the unmeasurable variables in the information vectors with their estimates. The proposed F-RLS algorithm has a 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

Signal processing, filtering and prediction, and parameter estimation have received much attention, e.g. [1–7]. For example, Ding and Chen established the multi-innovation identification theory and presented a multi-innovation stochastic gradient algorithm for linear regression models [8]. This multi-innovation parameter estimation method has been extended to pseudo-linear regression models [9] and used for self-tuning control [10].

This paper considers the parameter estimation problems, using the input–output data filtering, for the stochastic system with an autoregressive moving average (ARMA) disturbance, described by the controlled autoregressive autoregressive moving average (CARARMA) model [11,12], depicted in Fig. 1,

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

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 $e(t) := \frac{D(z)}{C(z)}v(t)$ is an ARMA model, $A(z)$, $B(z)$, $C(z)$ and $D(z)$ are polynomials in z^{-1}, and defined by

$$A(z) = 1 + a_1z^{-1} + a_2z^{-2} + \cdots + a_{n_a}z^{-n_a},$$
$$B(z) = b_1z^{-1} + b_2z^{-2} + \cdots + b_{n_b}z^{-n_b},$$
$$C(z) = 1 + c_1z^{-1} + c_2z^{-2} + \cdots + c_{n_c}z^{-n_c},$$

☆ This work was supported by the Shandong Province Colleges and Universities Outstanding Young Teachers in Domestic Visiting Scholars Project.
* Corresponding author.
E-mail addresses: dqwang64@163.com (D. Wang), fding@jiangnan.edu.cn (F. Ding).

1051-2004/$ – see front matter
doi:10.1016/j.dsp.2009.12.006

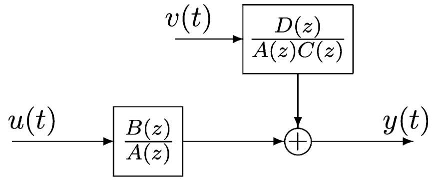

Fig. 1. The system described by CARARMA models.

$$D(z) = 1 + d_1 z^{-1} + d_2 z^{-2} + \cdots + d_{n_d} z^{-n_d}.$$

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

For special cases of the system in (1), many approaches can estimate their parameters. For example, when $C(z) = D(z)$, the system in (1) reduces to an equation error model, i.e., CAR model (Controlled Auto-Regression 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 [11,13–15]; when $C(z) = 1$, we get a CARMA model (controlled autoregressive moving average model), or called ARMAX model (autoregressive moving average model with exogenous input),

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

for which the recursive extended least squares algorithm or prediction error methods can identify its parameters a_i, b_i and d_i [11,13,16].

Although the instrumental variable least squares and bias compensation/correction least squares algorithms can identify the systems in (1) [11,17–21], the disadvantages are that they fail to obtain the parameter estimates of the noise models.

This paper discussed identification problems for CARARMA systems using the input–output data filtering technique. The objective is to present a filtering based recursive least squares algorithm (F-RLS) to estimate the system 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. The basic idea is to use the rational fraction transfer function $\frac{C(z)}{D(z)}$ to filter input–output data $\{u(t), y(t)\}$, resulting in an equation error (CAR or ARX) identification model and an ARMA noise identification model. Thus, we can estimate the parameters of both the system model $\frac{B(z)}{A(z)}$ and the noise model $\frac{D(z)}{C(z)}$ by replacing the unmeasurable variables in the information vectors with their estimates.

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

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

2. The RGELS algorithms

To show the advantages of the proposed F-RLS algorithm to be proposed later, the following gives the recursive generalized extended least squares algorithm for comparisons.

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

$$\boldsymbol{\theta} := \begin{bmatrix} \boldsymbol{\theta}_{\rm s} \\ \boldsymbol{\theta}_{\rm n} \end{bmatrix} \in \mathbb{R}^n,$$

$$\boldsymbol{\theta}_{\rm s} := [a_1, a_2, \ldots, a_{n_a}, b_1, b_2, \ldots, b_{n_b}]^{\rm T} \in \mathbb{R}^{n_a+n_b},$$

$$\boldsymbol{\theta}_{\rm n} := [c_1, c_2, \ldots, c_{n_c}, d_1, d_2, \ldots, d_{n_d}]^{\rm T} \in \mathbb{R}^{n_c+n_d},$$

$$\boldsymbol{\varphi}(t) = \begin{bmatrix} \boldsymbol{\varphi}_{\rm s}(t) \\ \boldsymbol{\varphi}_{\rm n}(t) \end{bmatrix} \in \mathbb{R}^n,$$

$$\boldsymbol{\varphi}_{\rm s}(t) := \left[-y(t-1), -y(t-2), \ldots, -y(t-n_a), u(t-1), u(t-2), \ldots, u(t-n_b)\right]^{\rm T} \in \mathbb{R}^{n_a+n_d},$$

$$\boldsymbol{\varphi}_{\rm n}(t) := \left[-e(t-1), -e(t-2), \ldots, -e(t-n_c), v(t-1), v(t-2), \ldots, v(t-n_d)\right]^{\rm T} \in \mathbb{R}^{n_c+n_d},$$

and the inner variables,

$$e(t) := \frac{D(z)}{C(z)} v(t) \tag{2}$$

or

$$e(t) = \left[1 - C(z)\right]e(t) + D(z)v(t) = \boldsymbol{\varphi}_{\rm n}^{\rm T}(t)\boldsymbol{\theta}_{\rm n} + v(t). \tag{3}$$

Here subscripts s and n denote the first letters of the words 'system' and 'noise', respectively. Using (2) and (3), Eq. (1) can be rewritten as

$$\begin{aligned} y(t) &= \left[1 - A(z)\right]y(t) + B(z)u(t) + e(t) \\ &= \boldsymbol{\varphi}_{\rm s}^{\rm T}(t)\boldsymbol{\theta}_{\rm s} + e(t) \qquad (4) \\ &= \boldsymbol{\varphi}_{\rm s}^{\rm T}(t)\boldsymbol{\theta}_{\rm s} + \boldsymbol{\varphi}_{\rm n}^{\rm T}(t)\boldsymbol{\theta}_{\rm n} + v(t) \\ &= \boldsymbol{\varphi}^{\rm T}(t)\boldsymbol{\theta} + v(t). \qquad (5) \end{aligned}$$

Because the information vector $\boldsymbol{\varphi}_{\rm n}(t)$ in $\boldsymbol{\varphi}(t)$ on the right-hand side contains unknown inner variables $e(t-i)$ and unmeasurable noise terms $v(t-i)$, the following standard recursive least squares algorithm cannot generate the estimate of the parameter vector $\boldsymbol{\theta}$ [11–13]:

$$\hat{\boldsymbol{\theta}}(t) = \hat{\boldsymbol{\theta}}(t-1) + \boldsymbol{L}(t)\left[y(t) - \boldsymbol{\varphi}^{\rm T}(t)\hat{\boldsymbol{\theta}}(t-1)\right], \tag{6}$$

$$\boldsymbol{L}(t) = \frac{\boldsymbol{P}(t-1)\boldsymbol{\varphi}(t)}{1 + \boldsymbol{\varphi}^{\rm T}(t)\boldsymbol{P}(t-1)\boldsymbol{\varphi}(t)}, \tag{7}$$

$$\boldsymbol{P}(t) = \left[\boldsymbol{I} - \boldsymbol{L}(t)\boldsymbol{\varphi}^{\rm T}(t)\right]\boldsymbol{P}(t-1). \tag{8}$$

The solution is to replace these unmeasurable variables $e(t-i)$ and $v(t-i)$ in $\boldsymbol{\varphi}_{\rm n}(t)$ of $\boldsymbol{\varphi}(t)$ with their estimates $\hat{e}(t-i)$ and $\hat{v}(t-i)$, respectively, and define

$$\hat{\boldsymbol{\varphi}}_{\rm n}(t) := \left[-\hat{e}(t-1), -\hat{e}(t-2), \ldots, -\hat{e}(t-n_c), \hat{v}(t-1), \hat{v}(t-2), \ldots, \hat{v}(t-n_d)\right]^{\rm T} \in \mathbb{R}^{n_c+n_d},$$

$$\hat{\boldsymbol{\varphi}}(t) := \begin{bmatrix} \boldsymbol{\varphi}_{\rm s}(t) \\ \hat{\boldsymbol{\varphi}}_{\rm n}(t) \end{bmatrix}.$$

Let $\hat{\boldsymbol{\theta}}(t) = \left[\begin{smallmatrix} \hat{\boldsymbol{\theta}}_{\rm s}(t) \\ \hat{\boldsymbol{\theta}}_{\rm n}(t) \end{smallmatrix}\right]$ be the estimate of $\boldsymbol{\theta} = \left[\begin{smallmatrix} \boldsymbol{\theta}_{\rm s} \\ \boldsymbol{\theta}_{\rm n} \end{smallmatrix}\right]$. Replacing $\boldsymbol{\theta}_{\rm s}$ with $\hat{\boldsymbol{\theta}}_{\rm s}(t)$ in (4), the estimate $\hat{e}(t)$ can be computed by

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

Replacing $\boldsymbol{\varphi}(t)$ and $\boldsymbol{\theta}$ in (5) with $\hat{\boldsymbol{\varphi}}(t)$ and $\hat{\boldsymbol{\theta}}(t)$, respectively, the estimate $\hat{v}(t)$ can be computed by

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

Note that $\hat{\boldsymbol{\varphi}}(t)$ is known at time t. Replacing $\boldsymbol{\varphi}(t)$ in (6)–(8) with $\hat{\boldsymbol{\varphi}}(t)$ yields a recursive generalized extended least squares algorithm (RGELS) for identifying the parameters of the CARARMA model in (5) [22,23]:

$$\hat{\boldsymbol{\theta}}(t) = \hat{\boldsymbol{\theta}}(t-1) + \boldsymbol{L}(t)\left[y(t) - \hat{\boldsymbol{\varphi}}^{\rm T}(t)\hat{\boldsymbol{\theta}}(t-1)\right], \tag{9}$$

$$\boldsymbol{L}(t) = \frac{\boldsymbol{P}(t-1)\hat{\boldsymbol{\varphi}}(t)}{1 + \hat{\boldsymbol{\varphi}}^{\rm T}(t)\boldsymbol{P}(t-1)\hat{\boldsymbol{\varphi}}(t)}, \tag{10}$$

$$\boldsymbol{P}(t) = \left[\boldsymbol{I} - \boldsymbol{L}(t)\hat{\boldsymbol{\varphi}}^{\rm T}(t)\right]\boldsymbol{P}(t-1), \qquad \boldsymbol{P}(0) = p_0\boldsymbol{I}_n, \tag{11}$$

$$\hat{\boldsymbol{\theta}}(t) = \begin{bmatrix} \hat{\boldsymbol{\theta}}_{\rm s}(t) \\ \hat{\boldsymbol{\theta}}_{\rm n}(t) \end{bmatrix}, \qquad \hat{\boldsymbol{\varphi}}(t) = \begin{bmatrix} \boldsymbol{\varphi}_{\rm s}(t) \\ \hat{\boldsymbol{\varphi}}_{\rm n}(t) \end{bmatrix}, \tag{12}$$

$$\boldsymbol{\varphi}_{\rm s}(t) = \left[-y(t-1), -y(t-2), \ldots, -y(t-n_a), u(t-1), u(t-2), \ldots, u(t-n_b)\right]^{\rm T}, \tag{13}$$

$$\hat{\boldsymbol{\varphi}}_{\rm n}(t) = \left[-\hat{e}(t-1), -\hat{e}(t-2), \ldots, -\hat{e}(t-n_c), \hat{v}(t-1), \hat{v}(t-2), \ldots, \hat{v}(t-n_d)\right]^{\rm T}, \tag{14}$$

$$\hat{e}(t) = y(t) - \boldsymbol{\varphi}_{\rm s}^{\rm T}(t)\hat{\boldsymbol{\theta}}_{\rm s}(t), \tag{15}$$

$$\hat{v}(t) = y(t) - \hat{\boldsymbol{\varphi}}^{\rm T}(t)\hat{\boldsymbol{\theta}}(t), \quad \text{or } \hat{v}(t) = \hat{e}(t) - \hat{\boldsymbol{\varphi}}_{\rm n}^{\rm T}(t)\hat{\boldsymbol{\theta}}_{\rm n}(t). \tag{16}$$

The initial values of the RGELS algorithm are generally taken as $\hat{\boldsymbol{\theta}}(0) =$ a small real vector or $\hat{\boldsymbol{\theta}}(0) = \mathbf{1}_n/p_0$ with $\mathbf{1}_n$ being an n-dimensional column vector whose element are all 1 and $p_0 = 10^6$, $\boldsymbol{P}(0) = \mathrm{diag}[\boldsymbol{P}_{\rm s}(0), \boldsymbol{P}_{\rm n}(0)]$, $\boldsymbol{P}_{\rm s}(0) = p_0\boldsymbol{I}_{n_a+n_b}$ and $\boldsymbol{P}_{\rm n}(0) = p_n\boldsymbol{I}_{n_c+n_d}$ with $\boldsymbol{I}_n$ being an identity matrix of size $n \times n$, $0 < p_n \leqslant 1$ or $p_n = 1$.

3. The filtering based recursive least squares algorithm

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

For the CARARMA 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

$$u_{\mathrm{f}}(t) := \frac{C(z)}{D(z)}u(t), \qquad y_{\mathrm{f}}(t) := \frac{C(z)}{D(z)}y(t), \tag{17}$$

$$\boldsymbol{\varphi}_{\mathrm{f}}(t) := \left[-y_{\mathrm{f}}(t-1), -y_{\mathrm{f}}(t-2), \ldots, -y_{\mathrm{f}}(t-n_a), u_{\mathrm{f}}(t-1), u_{\mathrm{f}}(t-2), \ldots, u_{\mathrm{f}}(t-n_b)\right]^{\mathrm{T}} \in \mathbb{R}^{n_a+n_b}. \tag{18}$$

Multiplying both sides of (1) by $\frac{C(z)}{D(z)}$ gives

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

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,

$$y_{\mathrm{f}}(t) = \left[1 - A(z)\right]y_{\mathrm{f}}(t) + B(z)u_{\mathrm{f}}(t) + v(t) = \boldsymbol{\varphi}_{\mathrm{f}}^{\mathrm{T}}(t)\boldsymbol{\theta}_{\mathrm{s}} + v(t). \tag{19}$$

Like (2), define the inner variable:

$$e(t) := \frac{D(z)}{C(z)}v(t) \tag{20}$$

or

$$e(t) = \boldsymbol{\varphi}_{\mathrm{n}}^{\mathrm{T}}(t)\boldsymbol{\theta}_{\mathrm{n}} + v(t). \tag{21}$$

For two identification models (19) and (21), 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 $\boldsymbol{\theta}$,

$$\hat{\boldsymbol{\theta}}_{\mathrm{s}}(t) = \hat{\boldsymbol{\theta}}_{\mathrm{s}}(t-1) + \boldsymbol{L}_{\mathrm{f}}(t)\left[y_{\mathrm{f}}(t) - \boldsymbol{\varphi}_{\mathrm{f}}^{\mathrm{T}}(t)\hat{\boldsymbol{\theta}}_{\mathrm{s}}(t-1)\right], \tag{22}$$

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

$$\boldsymbol{P}_{\mathrm{f}}(t) = \left[\boldsymbol{I} - \boldsymbol{L}_{\mathrm{f}}(t)\boldsymbol{\varphi}_{\mathrm{f}}^{\mathrm{T}}(t)\right]\boldsymbol{P}_{\mathrm{f}}(t-1), \tag{24}$$

$$\hat{\boldsymbol{\theta}}_{\mathrm{n}}(t) = \hat{\boldsymbol{\theta}}_{\mathrm{n}}(t-1) + \boldsymbol{L}_{\mathrm{n}}(t)\left[e(t) - \boldsymbol{\varphi}_{\mathrm{n}}^{\mathrm{T}}(t)\hat{\boldsymbol{\theta}}_{\mathrm{n}}(t-1)\right], \tag{25}$$

$$\boldsymbol{L}_{\mathrm{n}}(t) = \frac{\boldsymbol{P}_{\mathrm{n}}(t-1)\boldsymbol{\varphi}_{\mathrm{n}}(t)}{1 + \boldsymbol{\varphi}_{\mathrm{n}}^{\mathrm{T}}(t)\boldsymbol{P}_{\mathrm{n}}(t-1)\boldsymbol{\varphi}_{\mathrm{n}}(t)}, \tag{26}$$

$$\boldsymbol{P}_{\mathrm{n}}(t) = \left[\boldsymbol{I} - \boldsymbol{L}_{\mathrm{n}}(t)\boldsymbol{\varphi}_{\mathrm{n}}^{\mathrm{T}}(t)\right]\boldsymbol{P}_{\mathrm{n}}(t-1). \tag{27}$$

Because polynomials $C(z)$ and $D(z)$ are unknown, so are $u_{\mathrm{f}}(t)$ and $y_{\mathrm{f}}(t)$, the information vectors $\boldsymbol{\varphi}_{\mathrm{f}}(t)$ and $\boldsymbol{\varphi}_{\mathrm{n}}(t)$ are unknown, the algorithms in (22)–(27) are impossible to implement. Here, we still adopt the idea of replacing the unknown variables with their estimates to derive the F-RLS identification algorithms.

Substituting (20) into (1) gives

$$e(t) = A(z)y(t) - B(z)u(t) = y(t) - \boldsymbol{\varphi}_{\mathrm{s}}^{\mathrm{T}}(t)\boldsymbol{\theta}_{\mathrm{s}}. \tag{28}$$

From the above equation and (21), we have

$$y(t) = \boldsymbol{\varphi}_{\mathrm{s}}^{\mathrm{T}}(t)\boldsymbol{\theta}_{\mathrm{s}} + e(t) = \boldsymbol{\varphi}^{\mathrm{T}}(t)\boldsymbol{\theta} + v(t). \tag{29}$$

Replacing the unknown $\boldsymbol{\theta}_{\mathrm{s}}$ on the right-hand side of (28) with the estimate $\hat{\boldsymbol{\theta}}_{\mathrm{s}}(t-1)$, the estimate $\hat{e}(t)$ can be computed by

$$\hat{e}(t) = y(t) - \boldsymbol{\varphi}_{\mathrm{s}}^{\mathrm{T}}(t)\hat{\boldsymbol{\theta}}_{\mathrm{s}}(t-1).$$

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

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

From (21), 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 $\hat{v}(t)$ 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,

$$\hat{\boldsymbol{\theta}}_{\mathrm{n}}(t)=\left[\hat{c}_1(t),\hat{c}_2(t),\ldots,\hat{c}_{n_c}(t),\hat{d}_1(t),\hat{d}_2(t),\ldots,\hat{d}_{n_d}(t)\right]^{\mathrm{T}}$$

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

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

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

$$\hat{u}_{\mathrm{f}}(t)=\frac{\hat{C}(t,z)}{\hat{D}(t,z)}u(t),\qquad \hat{y}_{\mathrm{f}}(t)=\frac{\hat{C}(t,z)}{\hat{D}(t,z)}y(t)$$

or

$$\hat{D}(t,z)\hat{u}_{\mathrm{f}}(t)=\hat{C}(t,z)u(t),\qquad \hat{D}(t,z)\hat{y}_{\mathrm{f}}(t)=\hat{C}(t,z)y(t).$$

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

$$\begin{aligned}
\hat{u}_{\mathrm{f}}(t)&=\left[1-\hat{D}(t,z)\right]\hat{u}_{\mathrm{f}}(t)+\hat{C}(t,z)u(t)\\
&=-\hat{d}_1(t)\hat{u}_{\mathrm{f}}(t-1)-\hat{d}_2(t)\hat{u}_{\mathrm{f}}(t-2)-\cdots-\hat{d}_{n_d}(t)\hat{u}_{\mathrm{f}}(t-n_d)\\
&\quad+u(t)+\hat{c}_1(t)u(t-1)+\hat{c}_2(t)u(t-2)+\cdots+\hat{c}_{n_c}(t)u(t-n_c),\\
\hat{y}_{\mathrm{f}}(t)&=\left[1-\hat{D}(t,z)\right]\hat{y}_{\mathrm{f}}(t)+\hat{C}(t,z)y(t)\\
&=-\hat{d}_1(t)\hat{y}_{\mathrm{f}}(t-1)-\hat{d}_2(t)\hat{y}_{\mathrm{f}}(t-2)-\cdots-\hat{d}_{n_d}(t)\hat{y}_{\mathrm{f}}(t-n_d)\\
&\quad+y(t)+\hat{c}_1(t)y(t-1)+\hat{c}_2(t)y(t-2)+\cdots+\hat{c}_{n_c}(t)y(t-n_c).
\end{aligned}$$

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:

$$\hat{\boldsymbol{\varphi}}_{\mathrm{f}}(t)=\left[-\hat{y}_{\mathrm{f}}(t-1),-\hat{y}_{\mathrm{f}}(t-2),\ldots,-\hat{y}_{\mathrm{f}}(t-n_a),\hat{u}_{\mathrm{f}}(t-1),\hat{u}_{\mathrm{f}}(t-2),\ldots,\hat{u}_{\mathrm{f}}(t-n_b)\right]^{\mathrm{T}}\in\mathbb{R}^{n_a+n_b}.$$

Replacing the unknown information vector $\boldsymbol{\varphi}_{\mathrm{f}}(t)$ in (22)–(24) with $\hat{\boldsymbol{\varphi}}_{\mathrm{f}}(t)$, the unknown filtered output $y_{\mathrm{f}}(t)$ in (22) with $\hat{y}_{\mathrm{f}}(t)$, $\boldsymbol{\varphi}_{\mathrm{n}}(t)$ in (25)–(27) with $\hat{\boldsymbol{\varphi}}_{\mathrm{n}}(t)$, and the unknown variables $e(t)$ in (25) 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 CARARMA systems:

$$\hat{\boldsymbol{\theta}}_{\mathrm{s}}(t)=\hat{\boldsymbol{\theta}}_{\mathrm{s}}(t-1)+\boldsymbol{L}_{\mathrm{f}}(t)\left[\hat{y}_{\mathrm{f}}(t)-\hat{\boldsymbol{\varphi}}_{\mathrm{f}}^{\mathrm{T}}(t)\hat{\boldsymbol{\theta}}_{\mathrm{s}}(t-1)\right],\tag{30}$$

$$\boldsymbol{L}_{\mathrm{f}}(t)=\frac{\boldsymbol{P}_{\mathrm{f}}(t-1)\hat{\boldsymbol{\varphi}}_{\mathrm{f}}(t)}{1+\hat{\boldsymbol{\varphi}}_{\mathrm{f}}^{\mathrm{T}}(t)\boldsymbol{P}_{\mathrm{f}}(t-1)\hat{\boldsymbol{\varphi}}_{\mathrm{f}}(t)},\tag{31}$$

$$\boldsymbol{P}_{\mathrm{f}}(t)=\left[\boldsymbol{I}-\boldsymbol{L}_{\mathrm{f}}(t)\hat{\boldsymbol{\varphi}}_{\mathrm{f}}^{\mathrm{T}}(t)\right]\boldsymbol{P}_{\mathrm{f}}(t-1),\qquad \boldsymbol{P}_{\mathrm{f}}(0)=p_0\boldsymbol{I},\tag{32}$$

$$\hat{\boldsymbol{\varphi}}_{\mathrm{f}}(t)=\left[-\hat{y}_{\mathrm{f}}(t-1),-\hat{y}_{\mathrm{f}}(t-2),\ldots,-\hat{y}_{\mathrm{f}}(t-n_a),\hat{u}_{\mathrm{f}}(t-1),\hat{u}_{\mathrm{f}}(t-2),\ldots,\hat{u}_{\mathrm{f}}(t-n_b)\right]^{\mathrm{T}},\tag{33}$$

$$\begin{aligned}\hat{y}_{\mathrm{f}}(t)&=-\hat{d}_1(t)\hat{y}_{\mathrm{f}}(t-1)-\hat{d}_2(t)\hat{y}_{\mathrm{f}}(t-2)-\cdots-\hat{d}_{n_d}(t)\hat{y}_{\mathrm{f}}(t-n_d)\\
&\quad+y(t)+\hat{c}_1(t)y(t-1)+\hat{c}_2(t)y(t-2)+\cdots+\hat{c}_{n_c}(t)y(t-n_c),\end{aligned}\tag{34}$$

$$\begin{aligned}\hat{u}_{\mathrm{f}}(t)&=-\hat{d}_1(t)\hat{u}_{\mathrm{f}}(t-1)-\hat{d}_2(t)\hat{u}_{\mathrm{f}}(t-2)-\cdots-\hat{d}_{n_d}(t)\hat{u}_{\mathrm{f}}(t-n_d)\\
&\quad+u(t)+\hat{c}_1(t)u(t-1)+\hat{c}_2(t)u(t-2)+\cdots+\hat{c}_{n_c}(t)u(t-n_c),\end{aligned}\tag{35}$$

$$\hat{\boldsymbol{\theta}}_{\mathrm{n}}(t)=\hat{\boldsymbol{\theta}}_{\mathrm{n}}(t-1)+\boldsymbol{L}_{\mathrm{n}}(t)\left[\hat{e}(t)-\hat{\boldsymbol{\varphi}}_{\mathrm{n}}^{\mathrm{T}}(t)\hat{\boldsymbol{\theta}}_{\mathrm{n}}(t-1)\right],\tag{36}$$

$$\boldsymbol{L}_{\mathrm{n}}(t)=\frac{\boldsymbol{P}_{\mathrm{n}}(t-1)\hat{\boldsymbol{\varphi}}_{\mathrm{n}}(t)}{1+\hat{\boldsymbol{\varphi}}_{\mathrm{n}}^{\mathrm{T}}(t)\boldsymbol{P}_{\mathrm{n}}(t-1)\hat{\boldsymbol{\varphi}}_{\mathrm{n}}(t)},\tag{37}$$

$$\boldsymbol{P}_{\mathrm{n}}(t)=\left[\boldsymbol{I}-\boldsymbol{L}_{\mathrm{n}}(t)\hat{\boldsymbol{\varphi}}_{\mathrm{n}}^{\mathrm{T}}(t)\right]\boldsymbol{P}_{\mathrm{n}}(t-1),\qquad \boldsymbol{P}_{\mathrm{n}}(0)=p_0\boldsymbol{I},\tag{38}$$

$$\hat{\boldsymbol{\varphi}}_{\mathrm{n}}(t)=\left[-\hat{e}(t-1),\hat{e}(t-2),\ldots,-\hat{e}(t-n_c),\hat{v}(t-1),\hat{v}(t-2),\ldots,\hat{v}(t-n_d)\right]^{\mathrm{T}}, \tag{39}$$

$$\hat{e}(t)=y(t)-\boldsymbol{\varphi}_{\mathrm{s}}^{\mathrm{T}}(t)\hat{\boldsymbol{\theta}}_{\mathrm{s}}(t-1), \tag{40}$$

$$\hat{v}(t)=\hat{e}(t)-\hat{\boldsymbol{\varphi}}_{\mathrm{n}}^{\mathrm{T}}(t)\hat{\boldsymbol{\theta}}_{\mathrm{n}}(t), \tag{41}$$

$$\boldsymbol{\varphi}_{\mathrm{s}}(t)=\left[-y(t-1),-y(t-2),\ldots,-y(t-n_a),u(t-1),u(t-2),\ldots,u(t-n_b)\right]^{\mathrm{T}}, \tag{42}$$

$$\hat{\boldsymbol{\theta}}_{\mathrm{s}}(t)=\left[\hat{a}_1(t),\hat{a}_2(t),\ldots,\hat{a}_{n_a}(t),\hat{b}_1(t),\hat{b}_2(t),\ldots,\hat{b}_{n_b}(t)\right]^{\mathrm{T}}, \tag{43}$$

$$\hat{\boldsymbol{\theta}}_{\mathrm{n}}(t)=\left[\hat{c}_1(t),\hat{c}_2(t),\ldots,\hat{c}_{n_c}(t),\hat{d}_1(t),\hat{d}_2(t),\ldots,\hat{d}_{n_d}(t)\right]^{\mathrm{T}}. \tag{44}$$

To initialize the F-RLS algorithm, we take

$$\hat{\boldsymbol{\theta}}_{\mathrm{s}}(i)=\mathbf{1}_{n_a+n_b}/p_0,\qquad \hat{\boldsymbol{\theta}}_{\mathrm{n}}(i)=\mathbf{1}_{n_c+n_d}/p_0,\quad i\leqslant 0, \tag{45}$$

$$\boldsymbol{P}_{\mathrm{f}}(0)=p_0\boldsymbol{I}_{n_a+n_b},\qquad \boldsymbol{P}_{\mathrm{n}}(0)=p_0\boldsymbol{I}_{n_c+n_d},\quad p_0=10^6. \tag{46}$$

The steps of computing the parameter estimation in the F-RLS algorithms are listed in the following:

1. Let $t=1$, set the initial values of the parameter estimation vectors and covariance matrices according to (45) and (46), and $\hat{y}_{\mathrm{f}}(i)=1/p_0$, $\hat{u}_{\mathrm{f}}(i)=1/p_0$, $\hat{e}(i)=1/p_0$, $\hat{v}(i)=1/p_0$ for $i\leqslant 0$.
2. Collect the input–output data $u(t)$ and $y(t)$, construct the information vectors $\boldsymbol{\varphi}_{\mathrm{s}}(t)$ by (42), $\hat{\boldsymbol{\varphi}}_{\mathrm{f}}(t)$ by (33) and $\hat{\boldsymbol{\varphi}}_{\mathrm{n}}(t)$ by (39).
3. Compute $\hat{e}(t)$ by (40), the gain vector $\boldsymbol{L}_{\mathrm{n}}(t)$ by (37) and the covariance matrix $\boldsymbol{P}_{\mathrm{n}}(t)$ by (38).
4. Update the parameter estimate $\hat{\boldsymbol{\theta}}_{\mathrm{n}}(t)$ by (36).
5. Compute $\hat{v}(t)$ by (41), $\hat{y}_{\mathrm{f}}(t)$ by (34) and $\hat{u}_{\mathrm{f}}(t)$ by (35).
6. Compute the gain vector $\boldsymbol{L}_{\mathrm{f}}(t)$ by (31) and the covariance matrix $\boldsymbol{P}_{\mathrm{f}}(t)$ by (32).
7. Update the parameter estimate $\hat{\boldsymbol{\theta}}_{\mathrm{s}}(t)$ by (30).
8. Increase t by 1, go to step 2.

4. Example

Consider the following stochastic system,

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

$$A(z)=1+a_1z^{-1}+a_2z^{-2}=1+0.23z^{-1}+0.90z^{-2},$$

$$B(z)=b_1z^{-1}+b_2z^{-2}=-0.85z^{-1}+0.60z^{-2},$$

$$C(z)=1+c_1z^{-1}=1+0.62z^{-1},$$

$$D(z)=1+d_1z^{-1}=1-0.36z^{-1},$$

$$\boldsymbol{\theta}=[a_1,a_2,b_1,b_2,c_1,d_1]^{\mathrm{T}}=[0.23,0.90,-0.85,0.60,0.62,-0.36]^{\mathrm{T}}.$$

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 $\sigma^2=0.20^2$ and $\sigma^2=0.60^2$, respectively, their corresponding noise-to-signal ratio are $\delta_{\mathrm{ns}}=19.50\%$ and $\delta_{\mathrm{ns}}=58.49\%$, respectively. Applying the RGELS and the F-RLS algorithms to estimate the parameters of this system, the parameter estimates and their errors are shown in Tables 1–2, and the estimation errors $\delta:=\|\hat{\boldsymbol{\theta}}(t)-\boldsymbol{\theta}\|/\|\boldsymbol{\theta}\|$ versus t are shown in Fig. 2 with $\sigma^2=0.20^2$.

From Tables 1–2 and Fig. 2, we can get the following conclusions:

- The parameter estimation errors become (generally) smaller and smaller with the data length t increasing. This shows that the proposed algorithm is effective.
- The accuracy of the parameter estimation of the F-RLS algorithm is higher than that of the RGELS algorithm. This shows the F-RLS algorithm has a good identification performance compared with the RGELS algorithm.
- The parameter estimates given by the F-RLS algorithm converge fast to their true values compared with the RGELS algorithm.
- The proposed F-RLS algorithm requires less computational load than the RGELS algorithm because the dimensions of the covariance matrices $\boldsymbol{P}_{\mathrm{f}}(t)$ and $\boldsymbol{P}_{\mathrm{n}}(t)$ in the F-RLS algorithm are smaller than those of the covariance matrix $\boldsymbol{P}(t)$ in the RGELS algorithm because of $\boldsymbol{P}_{\mathrm{f}}(t)\in\mathbb{R}^{(n_a+n_b)\times(n_a+n_b)}$, $\boldsymbol{P}_{\mathrm{n}}(t)\in\mathbb{R}^{(n_c+n_d)\times(n_c+n_d)}$ and $\boldsymbol{P}(t)\in\mathbb{R}^{(n_a+n_b+n_c+n_d)\times(n_a+n_b+n_c+n_d)}$.

Table 1
The parameter estimates and their errors ($\sigma^2 = 0.20^2$, $\delta_{ns} = 19.50\%$).

Algorithms	t	a_1	a_2	b_1	b_2	c_1	d_1	δ (%)
RGELS	100	0.20681	0.90133	−0.87115	0.65060	0.42339	−0.35367	13.10552
	200	0.21544	0.91067	−0.87002	0.64512	0.57027	−0.28405	6.68902
	500	0.22748	0.90699	−0.86046	0.61897	0.60592	−0.29853	4.27844
	1000	0.23317	0.90510	−0.85494	0.60195	0.57394	−0.34732	3.08892
	1500	0.23423	0.90405	−0.85616	0.60394	0.58846	−0.36457	2.11842
	2000	0.23389	0.90376	−0.85709	0.60171	0.60909	−0.35117	1.06597
	2500	0.23288	0.90314	−0.85544	0.60484	0.60198	−0.36940	1.40319
	3000	0.23238	0.90214	−0.85430	0.60272	0.60434	−0.36041	1.06973
F-RLS	100	0.22711	0.90056	−0.80884	0.57948	0.62835	−0.35434	3.00828
	200	0.22961	0.90137	−0.82808	0.58974	0.62597	−0.35202	1.67162
	500	0.22943	0.90112	−0.84162	0.59522	0.62820	−0.34924	1.06257
	1000	0.22997	0.90056	−0.84584	0.59623	0.61814	−0.35251	0.60948
	1500	0.23035	0.90072	−0.84738	0.59683	0.62168	−0.35792	0.31662
	2000	0.23031	0.90077	−0.84849	0.59755	0.62624	−0.35562	0.52208
	2500	0.23020	0.90079	−0.84861	0.59813	0.62302	−0.36091	0.25558
	3000	0.23004	0.90071	−0.84888	0.59836	0.62277	−0.35619	0.32894
True values		0.23000	0.90000	−0.85000	0.60000	0.62000	−0.36000	

Table 2
The parameter estimates and their errors ($\sigma^2 = 0.60^2$, $\delta_{ns} = 58.49\%$).

Algorithms	t	a_1	a_2	b_1	b_2	c_1	d_1	δ (%)
RGELS	100	0.13788	0.88782	−0.89660	0.75787	0.50690	−0.42345	14.61695
	200	0.18082	0.93053	−0.90758	0.72737	0.60961	−0.30970	10.18802
	500	0.22355	0.92216	−0.87895	0.65015	0.61066	−0.31076	5.10022
	1000	0.23666	0.91574	−0.86355	0.60404	0.57655	−0.34747	3.21170
	1500	0.24124	0.91378	−0.86782	0.60982	0.58890	−0.36223	2.63073
	2000	0.24113	0.91242	−0.87081	0.60273	0.60893	−0.34812	1.99840
	2500	0.23853	0.91017	−0.86594	0.61216	0.60227	−0.36654	1.95002
	3000	0.23734	0.90744	−0.86268	0.60594	0.60386	−0.35827	1.52062
F-RLS	100	0.20586	0.90085	−0.84371	0.59567	0.58096	−0.35888	2.96856
	200	0.21898	0.90399	−0.85982	0.62929	0.65062	−0.30602	4.48324
	500	0.22279	0.90425	−0.85667	0.60706	0.65573	−0.31823	3.59926
	1000	0.22773	0.90322	−0.85325	0.59103	0.61114	−0.35677	0.89158
	1500	0.23215	0.90450	−0.85435	0.58846	0.61886	−0.37144	1.12110
	2000	0.23210	0.90511	−0.85713	0.59077	0.63232	−0.35686	1.15508
	2500	0.23117	0.90565	−0.85421	0.59363	0.62429	−0.36913	0.88664
	3000	0.22985	0.90519	−0.85382	0.59409	0.62526	−0.35779	0.66583
True values		0.23000	0.90000	−0.85000	0.60000	0.62000	−0.36000	

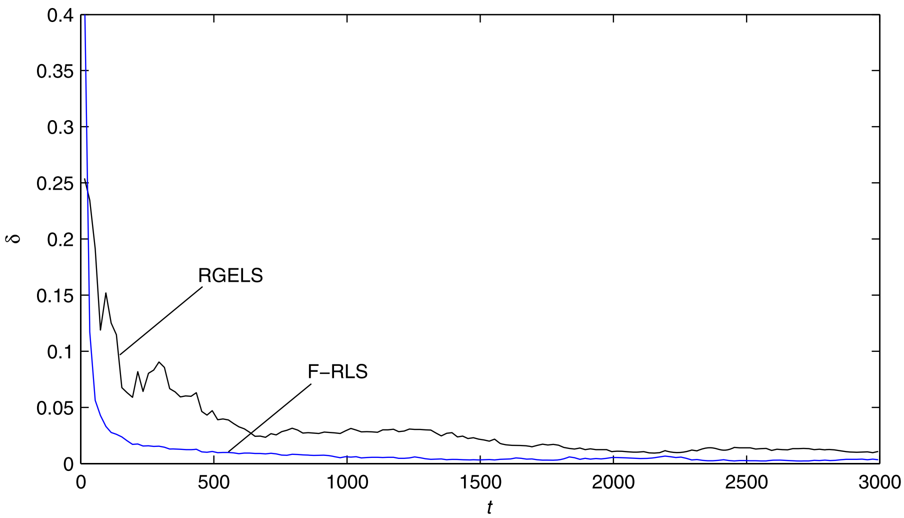

Fig. 2. The estimation errors δ versus t ($\sigma^2 = 0.20^2$).

5. Conclusions

A filtering based recursive least squares algorithm for a CARARMA systems is derived by filtering the input–output data with the estimated transfer function of the noise model. The proposed algorithms can require less computation and give highly accurate parameter estimates compared with the recursive generalized extended least squares algorithms. The proposed method can be extended to non-uniformly sampled systems [24,25] and nonlinear systems [26–28], and can be applied to estimate system parameters as the basis of designing filters or feedback control laws for uncertain systems or multirate systems [29–33]. The convergence analysis of the proposed F-RLS algorithm is difficult and requires further studies.

References

[1] M.B. Malik, M. Salman, State–space least mean square, Digital Signal Processing 18 (3) (2008) 334–345.
[2] R. Abrahamsson, S.M. Kay, P. Stoica, Estimation of the parameters of a bilinear model with applications to submarine detection and system identification, Digital Signal Processing 17 (4) (2007) 756–773.
[3] A. Zerguine, Convergence and steady-state analysis of the normalized least mean fourth algorithm, Digital Signal Processing 17 (1) (2007) 17–31.
[4] D. Zazula, A common approach to the analysis of cumulant-based AR and ARMA identification, Digital Signal Processing 12 (2) (2003) 233–251.
[5] C. James, Total least squares, matrix enhancement, and signal processing, Digital Signal Processing 4 (1) (1994) 21–39.
[6] L.L. Han, F. Ding, Multi-innovation stochastic gradient algorithms for multi-input multi-output systems, Digital Signal Processing 19 (4) (2009) 545–554.
[7] P. Löhnberg, G.H.J. Wisselink, Iterative least squares parameter estimation for ARMA pulse response and output disturbance, IEEE Transactions on Automatic Control 27 (6) (1982) 1252–1255.
[8] F. Ding, T. Chen, Performance analysis of multi-innovation gradient type identification methods, Automatica 43 (1) (2007) 1–14.
[9] F. Ding, P.X. Liu, G. Liu, Auxiliary model based multi-innovation extended stochastic gradient parameter estimation with colored measurement noises, Signal Processing 89 (10) (2009) 1883–1890.
[10] J.B. Zhang, F. Ding, Y. Shi, Self-tuning control based on multi-innovation stochastic gradient parameter estimation, Systems & Control Letters 58 (1) (2009) 69–75.
[11] L. Ljung, System Identification: Theory for the User, second ed., Prentice-Hall, Englewood Cliffs, NJ, 1999.
[12] F. Ding, Y.S. Xiao, A finite-data-window least squares algorithm with a forgetting factor for dynamical modeling, Applied Mathematics and Computation 186 (1) (2007) 184–192.
[13] G.C. Goodwin, K.S. Sin, Adaptive Filtering Prediction and Control, Prentice-Hall, Englewood Cliffs, NJ, 1984.
[14] F. Ding, P.X. Liu, H.Z. Yang, Parameter identification and intersample output estimation for dual-rate systems, IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans 38 (4) (2008) 966–975.
[15] Y.S. Xiao, F. Ding, Y. Zhou, M. Li, J.Y. Dai, On consistency of recursive least squares identification algorithms for controlled auto-regression models, Applied Mathematical Modelling 32 (11) (2008) 2207–2215.
[16] J. Ding, F. Ding, The residual based extended least squares identification method for dual-rate systems, Computers & Mathematics with Applications 56 (6) (2008) 1479–1487.
[17] F. Ding, T. Chen, L. Qiu, Bias compensation based recursive least squares identification algorithm for MISO systems, IEEE Transactions on Circuits and Systems – II: Express Briefs 53 (5) (2006) 349–353.
[18] W.X. Zheng, A bias-correction method for indirect identification of closed-loop systems, Automatica 31 (7) (1995) 1019–1024.
[19] W.X. Zheng, Least-squares identification of a class of multivariable systems with correlated disturbances, Journal of the Franklin Institute 336 (8) (1999) 1309–1324.
[20] W.X. Zheng, A bias correction method for identification of linear dynamic errors-in-variables models, IEEE Transactions on Automatic Control 47 (7) (2002) 1142–1147.
[21] 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 50 (9) (2003) 1232–1238.
[22] F. Ding, A recursive generalized extended least squares identification algorithm for Box–Jenkins models, Control and Decision 5 (6) (1990) 53–56 (in Chinese).
[23] X.M. Xie, F. Ding, Adaptive Control Systems, Tsinghua University Press, Beijing, 2002 (in Chinese).
[24] F. Ding, L. Qiu, T. Chen, Reconstruction of continuous-time systems from their non-uniformly sampled discrete-time systems, Automatica 45 (2) (2009) 324–332.
[25] Y.J. Liu, L. Xie, F. Ding, An auxiliary model based recursive least squares parameter estimation algorithm for non-uniformly sampled multirate systems, in: Proceedings of the Institution of Mechanical Engineers, Part I, Journal of Systems and Control Engineering 223 (4) (2009) 445–454.
[26] F. Ding, T. Chen, Identification of Hammerstein nonlinear ARMAX systems, Automatica 41 (9) (2005) 1479–1480.
[27] F. Ding, Y. Shi, T. Chen, Auxiliary model based least-squares identification methods for Hammerstein output-error systems, Systems & Control Letters 56 (5) (2007) 373–380.
[28] D.Q. Wang, F. Ding, Extended stochastic gradient identification algorithms for Hammerstein–Wiener ARMAX systems, Computers & Mathematics with Applications 56 (12) (2008) 3157–3164.
[29] Y. Shi, B. Yu, Output feedback stabilization of networked control systems with random delays modeled by Markov chains, IEEE Transactions on Automatic Control 54 (7) (2009) 1668–1674.
[30] M. Yan, Y. Shi, Robust discrete-time sliding mode control for uncertain systems with time-varying state delay, IET Control Theory & Applications 2 (8) (2008) 662–674.
[31] B. Yu, Y. Shi, H. Huang, $l_2 - l_\infty$ filtering for multirate systems using lifted models, Circuits, Systems and Signal Processing 27 (5) (2008) 699–711.
[32] Y. Shi, F. Ding, T. Chen, Multirate crosstalk identification in xDSL systems, IEEE Transactions on Communications 54 (10) (2006) 1878–1886.
[33] Y. Shi, F. Ding, T. Chen, 2-Norm based recursive design of transmultiplexers with designable filter length, Circuits, Systems and Signal Processing 25 (4) (2006) 447–462.

Dongqing Wang was born in Shenyang, Liaoning Province, China. She received the B.Sc. and the M.Sc. degrees from the Department of Electrical Engineering, Shandong University of Technology (Jinan, China) in 1986 and 1988, respectively, and the Ph.D. degree from the College of Automation Engineering, Tianjin University (Tianjin, China) in 2006. She is with the College of Automation Engineering, Qingdao University (Qingdao, China). Her research interests are stochastic systems, system identification, process modeling and control.

Feng Ding was born in Guangshui, Hubei Province. He received the B.Sc. degree from the Hubei University of Technology (Wuhan, China) in 1984, and the M.Sc. and Ph.D. degrees in automatic control both from the Department of Automation, Tsinghua University in 1991 and 1994, respectively.

From 1984 to 1988, he was an Electrical Engineer at the Hubei Pharmaceutical Factory, Xiangfan, China. From 1994 to 2002, he was with the Department of Automation, Tsinghua University, Beijing, China and he was a Research Associate at the University of Alberta, Edmonton, Canada from 2002 to 2005.

He was a Visiting Professor in the Department of Systems and Computer Engineering, Carleton University, Ottawa, Canada from May to December 2008 and a Research Associate in the Department of Aerospace Engineering, Ryerson University, Toronto, Canada, from January to October 2009.

He has been a Professor in the School of Communication and Control Engineering, Jiangnan University, Wuxi, China since 2004. He is a Colleges and Universities "Blue Project" Middle-Aged Academic Leader (Jiangsu, China). His current research interests include model identification and adaptive control. He co-authored the book *Adaptive Control Systems* (Tsinghua University Press, Beijing, 2002), and published over 100 papers on modeling and identification as the first author.