An Efficient Recursive Identification Algorithm for ARMAX Model

Jia, Li-Juan
Department of Electronic Engineering, School of Information Science and Technology, Beijing Institute of Technology

Kanae, Shunshoku
Department of Electrical and Electronic Systems Engineering, Faculty of Information Science and Electrical Engineering, Kyushu University

Yang, Zi-Jiang
Department of Electrical and Electronic Systems Engineering, Faculty of Information Science and Electrical Engineering, Kyushu University

Wada, Kiyoshi
Department of Electrical and Electronic Systems Engineering, Faculty of Information Science and Electrical Engineering, Kyushu University

https://doi.org/10.15017/1517812

出版情報：九州大学大学院システム情報科学紀要. 12 (1), pp.21-25, 2007-03-26. 九州大学大学院システム情報科学研究院

九州大学大学院
システム情報科学紀要
第12巻 第1号 平成19年3月

Research Reports on Information Science and
Electrical Engineering of Kyushu University
Vol.12, No.1, March 2007

An Efficient Recursive Identification Algorithm for ARMAX Model

Li-Juan JIA*, Shunshoku KANAE**, Zi-Jiang YANG** and Kiyoshi WADA**

(Received December 20, 2006)

Abstract: In this paper the problem of ARMAX model identification is studied and an efficient recursive robust identification algorithm applicable to ARMAX model is proposed. When applying directly LS method to ARMAX model estimation, the asymptotical bias appears. In order to estimate the bias, inspiring from the bias compensation least squares (BCLS) algorithm that is proposed by authors, a set of auxiliary linear backward predictors are introduced. With the help of those found orthogonal properties, a solution to the estimate of the bias is built. Consequently the consistent estimate for ARMAX model can be obtained via compensating the estimated bias. Moreover in order to satisfy the need of on-line identification of ARMAX model, recursive processing of the proposed algorithm is given. Simulation results are presented to illustrate the effectiveness of the proposed algorithm.

Keywords: Recursive identification, ARMAX models, Bias compensation, Backward prediction error

1. Introduction

Methods of recursive identification deal with the problem of building mathematical models of signals and systems on-line, at the same time as data is being collected. ARMAX (Auto-Regressive Moving Average eXogenous) model was introduced into system identification in 1) and is since then a basic model. Any linear finite-order system with stationary disturbances having a rational spectral density can be described by ARMAX model. In this paper we shall consider the problem of ARMAX model identification from a point of view of recursive processing and then propose an efficient recursive robust identification algorithm applicable to ARMAX model.

The recursive least squares (RLS) identification is a well-known technique for on-line parameter estimation. Unfortunately, the LS estimate is biased when directly apply to ARMAX model identification. To avoid the bias many methods appeared and the typical ones of them are the prediction error (PE) method and the instrumental variables (IV) method. Using the PE method leads to a more complex method and requires a numerical optimization of a nonlinear function that depends on the recorded data.

On the other hand, there are some estimation methods via the bias-compensation principle appeared in the control and signal processing literature such as modified LS (MLS) method[2], correction LS (CLS) method[3], bias-eliminated LS (BELS) method[4], and bias compensated LS (BCLS) method[5]. The key to bias compensation method is how to estimate the asymptotical bias of the LS estimate. Some devices for estimation of bias have been proposed in these bias compensating methods.

Recently, BELS method has been developed to apply for consistent parameter estimation of ARMAX models[6]. In order to estimate the bias of LS estimate, in the BELS method, an auxiliary parameter vector is introduced to construct an augmented ARMAX model and the bias estimate is derived from this auxiliary augmented model. Being different from BELS method, in our procedure no augmented ARMAX model appears and it allows us to estimate fewer parameters than the BELS method. Moreover in this paper both the recursive processing and the batch-processing are given, but only the batch form of BELS method is described in 6).

In this paper, in order to estimate the bias, inspiring from the BCLS algorithm[4], a set of auxiliary estimates are introduced. The introduced auxiliary estimates are used to construct one-step to n-step backward least-squares predictors. Then the cross-correlation functions of LS error and backward prediction (BWP) errors can be obtained. Meantime three useful orthogonal properties are discovered. One is that ARMAX model has a special property that the stochastic disturbance is orthogonal to the

* Department of Electronic Engineering, School of Information Science and Technology, Beijing Institute of Technology, Beijing 100081, P.R. China

** Department of Electrical and Electronic Systems Engineering

subspace spanned by the outputs. Another one is the orthogonal property between the LS error and data vector. The third one is that the introduced backward prediction errors and input-output data vector (asymptotically) are uncorrelated, *i.e.*, orthogonal. As we shall see, they are very useful to us to obtain a consistent estimate for bias of LS estimate. With the help of these orthogonal properties, an solution to the estimate of the bias is built. Consequently the consistent estimate for ARMAX model can be obtained via compensating the estimated bias. The introduced auxiliary estimates are constructed in LS sense, so the recursive approaches to them are very easy to implement.

This paper is organized as follows. Problem statement and description of ARMAX model are presented in section 2. The proposed method and the recursive processing implementation are described in section 3. Section 4 illustrates simulation results and conclusions are drawn in section 5.

2. Problem Statement

A general single-input single-output (SISO) discrete-time system can be described by an ARMAX (Auto-Regressive Moving Average eXogenous) model

$$A(q^{-1})y(k) = B(q^{-1})u(k) + C(q^{-1})w(k) \tag{1}$$

where $u(k)$ is input, $y(k)$ is output, and q^{-1} is the backward shift operator, i.e., $q^{-1}u(k) = u(k-1)$,

$$\begin{aligned} A(q^{-1}) &= 1 + a_1 q^{-1} + \cdots + a_n q^{-n} \\ B(q^{-1}) &= b_1 q^{-1} + \cdots + b_m q^{-m} \\ C(q^{-1}) &= c_0 + c_1 q^{-1} + \cdots + c_n q^{-n}. \end{aligned}$$

Let

$$\begin{aligned} \boldsymbol{\theta} &= \left[a_1 \cdots a_n \ \ b_1 \ldots b_m\right]^T \\ \boldsymbol{y}_k &= \left[y(k-1) \cdots y(k-n)\right]^T \\ \boldsymbol{u}_k &= \left[u(k-1) \ldots u(k-m)\right]^T \\ \boldsymbol{\phi}_k^T &= \left[-\boldsymbol{y}_k^T \ \ \boldsymbol{u}_k^T\right] \end{aligned}$$

The ARMAX model can be further written as a linear regression form

$$y(k) = \boldsymbol{\phi}_k^T \boldsymbol{\theta} + e(k) \tag{2}$$

where

$$e(k) = C(q^{-1})w(k) \tag{3}$$

represents the stochastic noise acting on the system, and $w(k)$ stands for the source of the disturbance.

The following standard assumptions are made.

- (A1) $A(\cdot)$ has all zeros strictly outside the unit disc.
- (A2) The input $u(t)$ is stationary and persistently exciting of a sufficient order.
- (A3) The disturbance source $w(k)$ is a zero-mean white noise and statistically uncorrelated with $u(k)$.
- (A4) The orders of model (n, m) are known.

The problem under study is, using input-output data $\{u(k), y(k)\}$ to make a consistent estimation for the ARMAX model parameter $\boldsymbol{\theta}$.

Based on the least-squares criterion, the LS estimate $\widehat{\boldsymbol{\theta}}_{LS}$ can be obtained as

$$\begin{aligned} \widehat{\boldsymbol{\theta}}_{LS} &= \Big(\sum_{k=1}^{N} \boldsymbol{\phi}_k \boldsymbol{\phi}_k^T\Big)^{-1} \sum_{k=1}^{N} \boldsymbol{\phi}_k y(k) & (4) \\ &= \widehat{R}_{\phi\phi}^{-1} \widehat{\boldsymbol{r}}_{\phi y} & (5) \end{aligned}$$

where

$$\widehat{R}_{\phi\phi} = \frac{1}{N}\sum_{k=1}^{N} \boldsymbol{\phi}_k \boldsymbol{\phi}_k^T, \quad \widehat{\boldsymbol{r}}_{\phi y} = \frac{1}{N}\sum_{k=1}^{N} \boldsymbol{\phi}_k y(k).$$

and the LS error $\xi(k)$ is described as follows.

$$\begin{aligned} \xi(k) &= y(k) - \boldsymbol{\phi}_k^T \widehat{\boldsymbol{\theta}}_{LS} \\ &= \boldsymbol{\phi}_k^T(\boldsymbol{\theta} - \widehat{\boldsymbol{\theta}}_{LS}) + e(k) & (6) \end{aligned}$$

By eqn. (4), we found the orthogonal property between the LS error $\xi(k)$ and data vector $\boldsymbol{\phi}_k$ as

Property 1

$$\sum_{k=1}^{N} \boldsymbol{\phi}_k (y(k) - \boldsymbol{\phi}_k^T \widehat{\boldsymbol{\theta}}_{LS}) = \sum_{k=1}^{N} \boldsymbol{\phi}_k \xi(k) = \mathbf{o}$$

which will be useful for derivation of the proposed method.

Substituting eqn. (2) into eqn. (4) yields

$$\widehat{\boldsymbol{\theta}}_{LS} = \boldsymbol{\theta} + \Big(\sum_{k=1}^{N} \boldsymbol{\phi}_k \boldsymbol{\phi}_k^T\Big)^{-1} \sum_{k=1}^{N} \boldsymbol{\phi}_k e(k) \tag{7}$$

Taking probability limit, we have

$$\operatorname*{plim}_{N\to\infty} \widehat{\boldsymbol{\theta}}_{LS} = \boldsymbol{\theta} + E[\boldsymbol{\phi}_k \boldsymbol{\phi}_k^T]^{-1} E[\boldsymbol{\phi}_k e_k]$$

This equation indicates that LS method can not give consistent estimate for ARMAX model due to the asymptotical bias. The bias of LS estimate is

$$\boldsymbol{h} = \operatorname*{plim}_{N\to\infty} \widehat{\boldsymbol{\theta}}_{LS} - \boldsymbol{\theta} = E[\boldsymbol{\phi}_k \boldsymbol{\phi}_k^T]^{-1} E[\boldsymbol{\phi}_k e(k)].$$

3. Our Proposed Algorithm

3.1 Bias Compensation

Bias compensation principle proposed by Sagara and Wada[2)] can be described as

$$\widehat{\boldsymbol{\theta}}_{BCLS} = \widehat{\boldsymbol{\theta}}_{LS} - \widehat{\boldsymbol{h}}$$

It means that if the noise-induced bias in the LS parameter estimate is obtained, compensating the LS estimate for the estimated bias can generate consistent parameter estimate for ARMAX model.

By assumption (A3), the input $u(k)$ is orthogonal to the stochastic disturbance $e(k)$, the bias estimate becomes

$$\widehat{\boldsymbol{h}} = -\widehat{R}_{\phi\phi}^{-1} \begin{bmatrix} I_n \\ O \end{bmatrix} \widehat{\boldsymbol{r}}_{ye}.$$

where

$$\operatorname*{plim}_{N\to\infty} \widehat{\boldsymbol{r}}_{ye} = E[\boldsymbol{y}_k e(k)].$$

then the consistent BCLS estimate for ARMAX model can be obtained as

$$\widehat{\boldsymbol{\theta}}_{BCLS} = \widehat{\boldsymbol{\theta}}_{LS} + \widehat{R}_{\phi\phi}^{-1} \begin{bmatrix} I_n \\ O \end{bmatrix} \widehat{\boldsymbol{r}}_{ye} \tag{8}$$

It is clear that estimation of $\widehat{\boldsymbol{r}}_{ye}$ becomes the key to BCLS method and our purpose is to find an efficient approach for estimation of $\widehat{\boldsymbol{r}}_{ye}$.

3.2 Backward Predictor and Prediction Error

By assumption (A3), it is also known that ARMAX model has an important feature that the stochastic disturbance $e(k)$ is orthogonal to the subspace spanned by the outputs $y(k-n-1), y(k-n-2), \cdots, y(k-2n)$. This property can be described as

Property 2.

$$E[e(k)y(k-n-i)] = 0, \quad i = 1\cdots n.$$

i.e.

$$\operatorname*{plim}_{N\to\infty} \frac{1}{N}\sum_{k=1}^{N} e(k)y(k-n-i) = 0, \quad i = 1\cdots n.$$

In this paper, utilizing this special property of ARMAX model helps us to find an ingenious approach to solution of the key problem to estimate the bias resulted from LS estimate.

Introduce the auxiliary estimates $\widehat{\boldsymbol{\beta}}_i$ $(i = 1\cdots n)$ defined as

$$\widehat{\boldsymbol{\beta}}_i = \Big[\sum_{k=1}^{N} \boldsymbol{\phi}_k \boldsymbol{\phi}_k^T\Big]^{-1} \sum_{k=1}^{N} \boldsymbol{\phi}_k y(k-n-i) \tag{9}$$

to construct i-step backward predictor

$$\widehat{y}(k-n-i) = \boldsymbol{\phi}_k^T \widehat{\boldsymbol{\beta}}_i \tag{10}$$

and i-step backward prediction error

$$\begin{aligned}\epsilon_i(k) &= y(k-n-i) - \widehat{y}(k-n-i) \\ &= y(k-n-i) - \boldsymbol{\phi}_k^T \widehat{\boldsymbol{\beta}}_i.\end{aligned}$$

From eqn. (9), it follows that

$$\sum_{k=1}^{N} \boldsymbol{\phi}_k (y(k-n-i) - \boldsymbol{\phi}_k^T \widehat{\boldsymbol{\beta}}_i) = \mathbf{o} \tag{11}$$

which shows the orthogonal property between data vector $\boldsymbol{\phi}_k$ and the backward prediction error $\epsilon_i(k)$.

Property 3.

$$\operatorname*{plim}_{N\to\infty} \frac{1}{N}\sum_{k=1}^{N} \boldsymbol{\phi}_k \epsilon_i(k) = \mathbf{o} \quad (i = 1\cdots n).$$

3.3 Cross-Correlation Function of LS Error and BWP Errors

For clarity and simplicity of presentation, we consider the one-step prediction error.

Denote f_1 be a cross-correlation function of the LS error $\xi(t)$ and one-step backward prediction error $\epsilon_1(t)$ described as

$$\begin{aligned} f_1 &= \sum_{k=1}^{N} \xi(k)\epsilon_1(k) \\ &= \sum_{k=1}^{N} (y(k) - \boldsymbol{\phi}_k^T \widehat{\boldsymbol{\theta}}_{LS})(y(k-n-1) - \boldsymbol{\phi}_k^T \widehat{\boldsymbol{\beta}}_1)\end{aligned}$$

By eqn. (6) and the orthogonal property between data vector $\boldsymbol{\phi}_k$ and the backward prediction error $\epsilon_1(k)$, f_1 can be rewritten as

$$f_1 = \sum_{k=1}^{N} e(k)(y(k-n-1) - \boldsymbol{\phi}_k^T \widehat{\boldsymbol{\beta}}_1) \tag{12}$$

Taking probability limit, we have

$$\begin{aligned}\operatorname*{plim}_{N\to\infty} \frac{1}{N} f_1 &= E[e(k)y(k-n-1)] \\ &\quad - \operatorname*{plim}_{N\to\infty} \widehat{\boldsymbol{\beta}}_1^T \begin{bmatrix} I_n \\ \mathbf{o} \end{bmatrix} E[\boldsymbol{y}_k e(k)].\end{aligned}$$

From **Property 2**, it follows that

$$\operatorname*{plim}_{N\to\infty}\frac{1}{N}f_1=-\operatorname*{plim}_{N\to\infty}\widehat{\beta}_1^T\begin{bmatrix}I_n\\ \mathbf{o}\end{bmatrix}E[\boldsymbol{y}_k e(k)]. \quad (13)$$

It is known that computing of n-dimension vector $\widehat{\boldsymbol{r}}_{ye}$ is a problem involving linear simultaneous equations in n unknowns. Eqn. (13) is only one linear equation which describes the relationship between $\widehat{\boldsymbol{r}}_{ye}$ and cross-correlation function f_1. In order to obtain the estimate $\widehat{\boldsymbol{r}}_{ye}$, we need n linear simultaneous equations.

Define $f_i,(i=1\cdots n)$ be a cross-correlation function of the LS error $\xi(k)$ and i-step backward prediction error $\epsilon_i(k)$ described as

$$f_i=\sum_{k=1}^{N}\xi(k)\epsilon_i(k)=$$
$$\sum_{k=1}^{N}(y(k)-\boldsymbol{\phi}_k^T\widehat{\boldsymbol{\theta}}_{LS,N})(y(k-n-i)-\boldsymbol{\phi}_k^T\widehat{\boldsymbol{\beta}}_i),$$

Taking probability limit, we have

$$\operatorname*{plim}_{N\to\infty}\frac{1}{N}f_i=E[e(k)y(k-n-i)]$$
$$-\operatorname*{plim}_{N\to\infty}\widehat{\beta}_i^T\begin{bmatrix}I_n\\ \mathbf{o}\end{bmatrix}E[\boldsymbol{y}_k e(k)].$$

From **Property 2**, it follows that

$$\operatorname*{plim}_{N\to\infty}\frac{1}{N}f_i=-\operatorname*{plim}_{N\to\infty}\widehat{\beta}_i^T\begin{bmatrix}I_n\\ \mathbf{o}\end{bmatrix}E[\boldsymbol{y}_k e(k)]. \quad (14)$$

Let

$$\boldsymbol{f}^T=[f_1\cdots f_n],$$

we can summarize the above results be

$$\boldsymbol{f}=\sum_{k=1}^{N}\begin{bmatrix}y(k-n-1)-\widehat{\boldsymbol{\beta}}_1^T\boldsymbol{\phi}_k\\ y(k-n-2)-\widehat{\boldsymbol{\beta}}_2^T\boldsymbol{\phi}_k\\ \vdots\\ y(k-2n)-\widehat{\boldsymbol{\beta}}_n^T\boldsymbol{\phi}_k\end{bmatrix}e(k)$$
$$=\sum_{k=1}^{N}\boldsymbol{\rho}_k e(k)-\widehat{\mathcal{B}}\sum_{k=1}^{N}\boldsymbol{\phi}_k e(k)$$

and

$$\widehat{\boldsymbol{r}}_{ye}=\left(\widehat{\mathcal{B}}\begin{bmatrix}I_n\\ O\end{bmatrix}\right)^{-1}\frac{1}{N}\boldsymbol{f} \quad (15)$$

where

$$\boldsymbol{\rho}_k=\begin{bmatrix}y(k-n-1)\\ y(k-n-2)\\ \vdots\\ y(k-2n)\end{bmatrix},\quad \widehat{\mathcal{B}}=\begin{bmatrix}\widehat{\boldsymbol{\beta}}_1^T\\ \widehat{\boldsymbol{\beta}}_2^T\\ \vdots\\ \widehat{\boldsymbol{\beta}}_n^T\end{bmatrix}.$$

3.4 Computation of Cross-correlation Function

From the definition of vector $\boldsymbol{f}$, **Property 1** and **Property 3**, we can derive the two forms for computing the cross-correlation function of LS error and BWP errors.

$$\frac{1}{N}\boldsymbol{f}=\frac{1}{N}\sum_{k=1}^{N}\begin{bmatrix}y(k-n-1)\\ y(k-n-2)\\ \vdots\\ y(k-2n)\end{bmatrix}(y(k)-\boldsymbol{\phi}_k^T\widehat{\boldsymbol{\theta}}_{LS})$$
$$=\frac{1}{N}\sum_{k=1}^{N}\boldsymbol{\rho}_k y(k)-\frac{1}{N}\sum_{k=1}^{N}\boldsymbol{\rho}_k\boldsymbol{\phi}_k^T\widehat{\boldsymbol{\theta}}_{LS}$$
$$=\widehat{\boldsymbol{r}}_{\rho y}-\widehat{R}_{\phi\rho}^T\widehat{\boldsymbol{\theta}}_{LS} \quad (16)$$

or

$$\frac{1}{N}\boldsymbol{f}=\frac{1}{N}\sum_{k=1}^{N}\left\{\begin{bmatrix}y(k-n-1)\\ y(k-n-2)\\ \vdots\\ y(k-2n)\end{bmatrix}-\begin{bmatrix}\widehat{\boldsymbol{\beta}}_1^T\\ \widehat{\boldsymbol{\beta}}_2^T\\ \vdots\\ \widehat{\boldsymbol{\beta}}_n^T\end{bmatrix}\boldsymbol{\phi}_k\right\}y(k)$$
$$=\frac{1}{N}\sum_{k=1}^{N}\boldsymbol{\rho}_k y(k)-\widehat{\mathcal{B}}\frac{1}{N}\sum_{k=1}^{N}\boldsymbol{\phi}_k y(k)$$
$$=\widehat{\boldsymbol{r}}_{\rho y}-\widehat{\mathcal{B}}\widehat{\boldsymbol{r}}_{\phi y} \quad (17)$$

3.5 Recursive Implementation of the Proposed Algorithm

Based on the above discussions, the recursive form of the proposed algorithm is given as follows:

***Step* 1.** Calculate the LS estimate $\widehat{\boldsymbol{\theta}}_{LS,N}$ and the auxiliary estimate $\widehat{\mathcal{B}}_N$ by the conventional RLS algorithm:

$$\widehat{\boldsymbol{\theta}}_{LS,N}=\widehat{\boldsymbol{\theta}}_{LS,N-1}+\frac{P_{N-1}\boldsymbol{\phi}_N\left(y_N-\boldsymbol{\phi}_N^T\widehat{\boldsymbol{\theta}}_{LS,N-1}\right)}{1+\boldsymbol{\phi}_N^TP_{N-1}\boldsymbol{\phi}_N}$$

$$\widehat{\mathcal{B}}_N=\widehat{\mathcal{B}}_{N-1}+\frac{P_{N-1}\left(\boldsymbol{\rho}_N-\widehat{\mathcal{B}}_{N-1}\boldsymbol{\phi}_N\right)\boldsymbol{\phi}_N^T}{1+\boldsymbol{\phi}_N^TP_{N-1}\boldsymbol{\phi}_N}$$

$$P_N=P_{N-1}-\frac{P_{N-1}\boldsymbol{\phi}_N\boldsymbol{\phi}_N^TP_{N-1}}{1+\boldsymbol{\phi}_N^TP_{N-1}\boldsymbol{\phi}_N}$$

***Step* 2.** Calculate $\boldsymbol{f}_N$:

$$\begin{aligned}
\boldsymbol{r}_{\rho y,N} &= \boldsymbol{r}_{\rho y,N-1} + \boldsymbol{\rho}_N y_N \\
R_{\phi\rho,N} &= R_{\phi\rho,N-1} + \boldsymbol{\phi}_N \boldsymbol{\rho}_N^T \\
\boldsymbol{r}_{\phi y,N} &= \boldsymbol{r}_{\phi y,N-1} + \boldsymbol{\phi}_N y_N \\
\boldsymbol{f}_N &= \boldsymbol{r}_{\rho y,N} - R_{\phi\rho,N}^T \widehat{\boldsymbol{\theta}}_{LS,N} \quad \text{or} \\
&= \boldsymbol{r}_{\rho y,N} - \widehat{\mathcal{B}}_N \boldsymbol{r}_{\phi y,N}
\end{aligned}$$

***Step* 3.** Calculate $\widehat{\boldsymbol{r}}_{ye}$

$$\widehat{\boldsymbol{r}}_{ye} = (\widehat{\mathcal{B}}_N \begin{bmatrix} I_n \\ O \end{bmatrix})^{-1} \frac{\boldsymbol{f}_N}{N}$$

***Step* 4.** Calculate the consistent parameter estimate for ARMAX model via

$$\widehat{\boldsymbol{\theta}}_{BCLS,N} = \widehat{\boldsymbol{\theta}}_{LS,N} + P_N \begin{bmatrix} I_n \\ O \end{bmatrix} (\widehat{\mathcal{B}}_N \begin{bmatrix} I_n \\ O \end{bmatrix})^{-1} \boldsymbol{f}_N$$

***Step* 5.** N=N+1 and return to ***Step* 1** until convergence.
Initial values at $N = 0$ are given as follows:

$$\begin{aligned}
&P_0 = \alpha \boldsymbol{I}, \widehat{\boldsymbol{\theta}}_{LS,0} = \mathbf{o}, \widehat{\mathcal{B}}_0 = O, \\
&\boldsymbol{r}_{\rho y,0} = \mathbf{o}, \boldsymbol{r}_{\phi y,0} = \mathbf{o}, R_{\phi\rho,0} = O.
\end{aligned}$$

The batch form is often applied for theoretical analysis and discussion, so we also summarize the batch form of the proposed algorithm as follows.

$$\widehat{\boldsymbol{\theta}}_{BCLS} = \widehat{\boldsymbol{\theta}}_{LS} + \widehat{R}_{\phi\phi}^{-1} \begin{bmatrix} I_n \\ O \end{bmatrix} \widehat{\boldsymbol{r}}_{ye} \tag{18}$$

$$\widehat{\mathcal{B}} = \sum_{k=1}^{N} \begin{bmatrix} y(k-n-1) \\ y(k-n-2) \\ \vdots \\ y(k-2n) \end{bmatrix} \boldsymbol{\phi}_k^T \widehat{R}_{\phi\phi}^{-1} = \widehat{R}_{\phi\rho}^T \widehat{R}_{\phi\phi}^{-1} \tag{19}$$

$$\widehat{\boldsymbol{r}}_{ye} = \left(\widehat{\mathcal{B}} \begin{bmatrix} I_n \\ O \end{bmatrix}\right)^{-1} \left(\widehat{\boldsymbol{r}}_{\rho y} - \widehat{R}_{\phi\rho}^T \widehat{\boldsymbol{\theta}}_{LS}\right) \tag{20}$$

or

$$\widehat{\boldsymbol{r}}_{ye} = \left(\widehat{\mathcal{B}} \begin{bmatrix} I_n \\ O \end{bmatrix}\right)^{-1} \left(\widehat{\boldsymbol{r}}_{\rho y} - \widehat{\mathcal{B}} \widehat{\boldsymbol{r}}_{\phi y}\right) \tag{21}$$

4. Simulation Results

Consider a second-order ARMAX model given by

$$\begin{aligned}
A(q^{-1}) &= 1 - 1.5q^{-1} + 0.7q^{-2} \\
B(q^{-1}) &= 1.0q^{-1} + 0.5q^{-2} \\
C(q^{-1}) &= 1 + 1.0q^{-1} + 0.2q^{-2}.
\end{aligned}$$

The input $u(k)$ is taken as white noise with unit variance. The $w(k)$ is a white noise with zero mean and variance σ_w^2 is chosen as 0.81 and 1.44 that implies the high noise environments.

The mean values of parameter estimates obtained by RLS method and the proposed algorithm for 20 runs are listed in **Table 1**. Data length is chosen as 2000 and 5000 respectively. Simulation results demonstrate that the RLS method gives biased results, but the proposed algorithm can give consistent estimates via compensating the estimated bias. Moreover it is shown that estimation accuracy is satisfactory even in high noise environments and it implies that the proposed algorithm is efficient and robust.

Table 1 Simulation results.

	σ_w^2	N	a_1	a_2	b_1	b_2
LS	0.81	2000	-1.2634	0.4838	0.9865	0.7342
		5000	-1.2612	0.4847	1.0064	0.7373
BCLS	0.81	2000	-1.5003	0.7004	0.9878	0.4988
		5000	-1.5012	0.7008	1.0004	0.4994
LS	1.44	2000	-1.1483	0.3832	0.9965	0.8481
		5000	-1.1602	0.3939	1.0072	0.8361
BCLS	1.44	2000	-1.4903	0.6939	0.9915	0.5089
		5000	-1.5023	0.7013	1.0070	0.4919
True value			-1.5	0.7	1.0	0.5

5. Conclusions

In this paper we describe the efficient robust identification algorithm applicable to ARMAX model from a point view of recursive processing. The contribution of this paper is that a new approach to estimation for the bias of LS estimate is proposed. It is worthy to find some useful properties that the stochastic noise is orthogonal to the subspace spanned by the outputs, the LS error and the introduced backward prediction errors are also (asymptotically) orthogonal to input-output data vector. With the help of these orthogonal properties, the efficient algorithm has been established to provide consistent parameter estimation for ARMAX model. Due to the auxiliary estimates are introduced in LS sense, the recursive implementations for them are easy to be obtained. Moreover for the purpose of the theoretical analysis, the batch-processing form of the proposed method is also given. Simulations have been performed to verify the efficiency and robustness of the proposed algorithm.

References

1) K.J. Aström and T. Bohlin, *IFAC Symposium on Self-Adaptive Systems*, Teddington, England, 1965.
2) S. Sagara and K. Wada, *Int.J. Control*, **25**(3), 329-343, 1977.
3) P. Stoica and T. Söderström *Int.J. Control*, **35**, 449-457, 1982.
4) C. B. Feng and W. X. Zheng, *IEE Proc.-Control Theory and Applications*, **138**(5), 484-492, 1991.
5) L.J. Jia, M. Ikenoue, C.Z. Jin and K. Wada, *Proc. of the 40th IEEE Conference on Decision and Control*, 3332-3337, Orlando, Florida USA, 2001.
6) W.X. Zheng, *Proc. of 2002 IFAC 15th World Congress*, Barcelona, Spain, 2002.