Received: 7 May 2021 | Revised: 6 December 2021 | Accepted: 14 December 2021

DOI: 10.1002/rnc.5972

RESEARCH ARTICLE

WILEY

Augmented flexible least squares algorithm for time-varying parameter systems

Jing Chen[1] | **Liuxiao Guo**[1] | **Manfeng Hu**[1] | **Min Gan**[2] | **Quanmin Zhu**[3]

[1]School of Science, Jiangnan University, Wuxi, People's Republic of China

[2]College of Mathematics and Computer Science, Fuzhou University, Fuzhou, People's Republic of China

[3]Department of Engineering Design and Mathematics, University of the West of England, Bristol, UK

Correspondence
Jing Chen, School of Science, Jiangnan University, Wuxi 214122, People's Republic of China.
Email: chenjing1981929@126.com

Funding information
National Natural Science Foundation of China, Grant/Award Number: 61973137; Funds of the Science and Technology on Near-Surface Detection Laboratory, Grant/Award Numbers: 61424140202, 61424140207; Natural Science Foundation of Jiangsu Province, Grant/Award Number: BK20201339

Abstract

This study proposes an augmented flexible least squares (FLS) algorithm for time-varying parameter systems. The parameter estimates, obtained by minimizing the squared residual measurement and dynamic errors, can catch the true values through a penalized term/weight. The algorithm associated properties are analyzed accordingly. By absorbing all into time varying parameters, the algorithm can convert complex nonlinear processes into various linear relations in time varying parameters. Thus, it can be extended to many kinds of systems. Compared to the classical FLS algorithm, the algorithm proposed in this article has less computational efforts and concise structures. To show the effectiveness of the algorithm and help the readers to follow systematically, this study provides several simulation examples.

KEYWORDS

computational effort, filtered estimates, flexible least squares algorithm, smoothed estimates, time-varying parameter system

1 | INTRODUCTION

Many social and biological problems, for example, the money demand of a country, the number of an endangered species,[1-3] can be modeled by time-varying parameter systems. Once such models are established, some pertinent strategies based on the predictive models can be applied.[4-6] Therefore, in order to build a precise model and use it to obtain insights into the mechanisms and make predictions, system identification is of significant importance.[7-9] Over the years, a plethora of identification algorithms have been developed, including the gradient iterative (GI) algorithm,[10,11] the expectation maximization (EM) algorithm, and the least squares (LS) algorithm.[12,13] By using these algorithms, a dynamical system can be expressed by an approximated mathematical model obtainable from measured data.

Time-varying parameter system identification is more challenging when comparing with the constant system identification, because the parameters evolve unanticipatedly.[14-16] Recently, most of the identification methods for time-varying parameter systems usually have assumed that the parameters keep unchanged in a fixed interval and evolve quickly to another mode, and the number of the collected input-output data in the fixed interval is larger than that of the unknown parameters.[17,18] For example, Yang and Yin[19] proposed an EM algorithm for linear parameter varying dual-rate systems, the identity of the sub-model is estimated in the E-step, and the parameters of each sub-model are updated in the M-step based on the estimated identities. Lu et al.[20] presented a variational Bayesian algorithm for a time-varying parameter ARX model, while the identities and parameters of the ARX model are iteratively estimated. However, when the parameters are varying in each sampling instant, the above methods are unavailable.

To deal with the systems whose parameters evolve slowly in each sampling instant, the flexible least squares (FLS) algorithm is a good option. The FLS algorithm was first developed by Kalaba and Tesfatsion,[1] its basic idea is to construct a cost function which is constituted of two kinds of errors: the first is the error of all coefficient sequence estimates, and the second is the error between the true outputs and predicted outputs.[21,22] To get the FLS estimates, some extended matrices whose orders are larger than the number of the unknown parameters should be constructed, which leads to heavily demanded computational efforts and complex structures of the algorithm.

This article develops an augmented FLS algorithm for time-varying parameter systems. The proposed algorithm can estimate the parameters in each sampling instant with less computational efforts and simple structures. In addition, it can be extended to polynomial nonlinear models,[23] rational models,[24] switching models and exponential autoregressive models,[25,26] by replacing the unknown nonlinear structures with linear structures of time-varying parameters. Briefly, the contributions of this article are summarized as follows:

1. The augmented FLS algorithm has less computational efforts and concise structures.
2. The augmented FLS algorithm is applicable for many classes of systems.
3. The augmented FLS algorithm is a probability-free formulation which does not require the usual probabilistic assumptions.

For the remainder of this article, Section 2 explains the time-varying parameter model and its various representatives. Section 3 introduces the framework of the FLS algorithm. Section 4 derives the augmented FLS algorithm. Section 5 analyzes some properties of the augmented FLS algorithm. Section 6 provides several bench test simulation examples. Finally, Section 7 summarizes the study and discusses some future directions.

2 | TIME-VARYING PARAMETER SYSTEMS

In general, a linear time-varying system can be described as

$$o(k)=\sum_{i=1}^{n}a_io(k-i)+\sum_{j=1}^{m}b_j(k)u(k-j)+v(k),$$

where $u(k)$ and $o(k)$ are the input and output, respectively, $a_i(k), i=1,2,\dots,n$ and $b_j(k), j=1,2,\dots,m$ are the unknown time-varying parameters, $v(k)$ is a noise that satisfies $v(k)\sim N(0,\sigma^2)$. This structure has excellent extrapolation properties, for example, almost all classes of models can be regarded as a subset or a special case of the time-varying model.

2.1 | The time-varying model and its special cases

2.1.1 | Polynomial nonlinear model

The polynomial nonlinear model is widely studied in theory and applications,[27] and usually written by

$$o(k)=\sum_{i=1}^{n_a}a_io(k-i)+\sum_{j=1}^{n_b}b_jf(u(k-j))+v(k). \tag{1}$$

To estimate the parameters of the polynomial nonlinear model, the nonlinear structure $f(\cdot)$ is often assumed to be known in prior. In this article, we transform the polynomial nonlinear model into a time-varying model.

Let

$$\beta_j(k)=b_j\frac{f(u(k-j))}{u(k-j)}.$$

Then, it gives rise to

$$o(k)=\sum_{i=1}^{n_a}a_io(k-i)+\sum_{j=0}^{n_a}\beta_j(k)u(k-j)+v(k). \tag{2}$$

Remark 1. For the polynomial nonlinear model, the nonlinear structures are always assumed to be known in prior; Otherwise, one should determine the structures of the systems first, and then use the identification algorithms to get the parameter estimates based on the known structures.

2.1.2 | Rational model

The rational model is written as the ratio of two polynomial expressions,[28]

$$o(k) = \frac{f(k)}{g(k)} + v(k), \tag{3}$$

where $f(k)$ and $g(k)$ are expressed as

$$\begin{aligned} f(k) &= \boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{\theta}_a, \\ g(k) &= \boldsymbol{\psi}^{\mathrm{T}}(k)\boldsymbol{\theta}_b, \\ \boldsymbol{\varphi}(k) &= [\varphi_1(k), \dots, \varphi_n(k)]^{\mathrm{T}} \in \mathbb{R}^n, \\ \boldsymbol{\psi}(k) &= [\psi_1(k), \dots, \psi_m(k)]^{\mathrm{T}} \in \mathbb{R}^m, \end{aligned}$$

and $\varphi_i(k), i = 1, \dots, n$ and $\psi_j(k), j = 1, \dots, m$ are constituted of past inputs and outputs, and whose structures are known a priori, $\boldsymbol{\theta}_a$ and $\boldsymbol{\theta}_b$ are the unknown parameters to be estimated and can be expressed as $\boldsymbol{\theta}_a = [a_i]^{\mathrm{T}} \in \mathbb{R}^n, \boldsymbol{\theta}_b = [b_j] \in \mathbb{R}^m, i = 1, \dots, n, j = 1, \dots, m$.

Let

$$a_i(k) = \frac{a_i}{g(k)}.$$

The rational model can be expressed as the following time-varying system,

$$o(k) = \sum_{i=1}^{n} a_i(k)\varphi_i(k) + v(k).$$

Remark 2. Some of the entries in $\boldsymbol{\psi}(k)$ of (3) involve the current output $o(k)$, which is correlated with the noise $v(k)$. Take this into account, a bias compensation based identification algorithm has been applied to the rational models.[28,29] However, the noise estimates in the compensation term need to be updated in each iteration by using the parameter estimates. Thus, the bias compensation identification algorithm is sensitive to the noise estimates and has heavy computational efforts.

2.1.3 | Switching model

Switching systems, which have a number of modes with different dynamical properties, can be written as

$$o(k) = \boldsymbol{\psi}^{\mathrm{T}}(k)\boldsymbol{\theta}_i, \quad k \in (k_{i-1}, k_i],$$

and it can be turned into

$$o(k) = \boldsymbol{\psi}^{\mathrm{T}}(k)\boldsymbol{\theta}(k).$$

In the literature of switching model identification, the self-organizing map (SOM) method and the expectation maximization (EM) method are two efficient methods.[20,25] The SOM method can classify the data based on the similarity among the data.[30] When using the SOM method for the switching model, the residual errors of the sub-models should be computed in each iteration to determine which is the suitable model at the sampling instant k. The EM algorithm constitutes of two steps: E step and M step,[31] where the identities (latent variables) of each model are estimated in the E step, and then the parameters are obtained in the M step based on the estimated latent variables.

Remark 3. The SOM and EM methods for switching models need to compute the residual errors or the latent variables in each iteration, and both of them are off-line algorithms. Thus, they cannot update the estimates through new generated data.

2.1.4 | Exponential autoregressive model

In application, some phenomena essentially display unique nonlinear behaviors, which cannot be explained well by polynomial nonlinear model. For example, the exponential autoregressive (ExpAR) model, in which some parameters are involved in the nonlinear structures,[32] is defined as

$$o(k)=\sum_{i=1}^{n}(a_i+b_ie^{-ro^2(k-i)})o(k-i)+v(k).$$

Let

$$b_i(k)=a_i+b_ie^{-ro^2(k-1)},\quad i=1,\dots,n.$$

The ExpAR model is simplified as a time-varying parameter model

$$o(k)=\sum_{i=1}^{n}b_i(k)o(k-i)+v(k). \tag{4}$$

Researchers have proposed numerous identification methods for ExpAR models, such as the variable projection algorithms, the hierarchical on-line algorithms.[2,26] In these methods, the hidden variables are usually replaced by their estimates.

Remark 4. For the ExpAR model in (4) and the identification methods in References 2 and 26, the exponential coefficient r is estimated interactively with the system parameters $a_i, b_i, i=1,\dots,n$. Once r has a poor estimation accuracy, the algorithms may have slow convergence rates or even be divergent.

2.2 | The identification model

The linear time-varying system in (1) can be modeled by the following regression vector form,

$$o(k)=\boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{\theta}(k)+v(k), \tag{5}$$

where

$$\boldsymbol{\varphi}(k)=[o(k-1),\dots,o(k-n),u(k-1),\dots,u(k-m)]^{\mathrm{T}}, \tag{6}$$

$$\boldsymbol{\theta}(k)=[a_1(k),\dots,a_n(k),b_1(k),\dots,b_m(k)]^{\mathrm{T}}. \tag{7}$$

Collect the input-output data for $k=1,2,\dots,L$ and define

$$\mathbf{O}(L)=[o(L),o(L-1),\dots,o(1)]^{\mathrm{T}}\in\mathbb{R}^{L},$$
$$\mathbf{V}(L)=[v(L),v(L-1),\dots,v(1)]^{\mathrm{T}}\in\mathbb{R}^{L}.$$

The purpose of this article is to propose an algorithm to estimate the unknown time-varying parameters by using the collected input and output data, and the proposed algorithm has the following features:

1. The nonlinear structures of the model do not require to be known in prior.
2. The proposed algorithm is an on-line algorithm which has less computational efforts and concise structures.
3. The proposed algorithm does not need to compute the residual errors and the latent variables at each sampling instant.
4. There is no unknown variable in the information vector when using the proposed algorithm for the time-varying parameter systems.

Remark 5. Compared with the LPV identification algorithm,[33,34] the algorithm proposed in this article can deal with time-varying parameter systems whose parameters are changing in each sampling instant.

3 | THE FLS FRAMEWORK

In Reference 1, an FLS algorithm was proposed for a time-series linear regression model with the assumption that the parameters of the model evolve only slowly over time. Two model specification errors are considered in each sampling instant to estimate the current parameters: one is the residual measurement error given by the discrepancy between the output and the regression model, and the other is the residual dynamic error between the neighboring parameter vector estimates. To help the readers to follow the augmented FLS algorithm, the framework of the classical FLS algorithm is introduced in this section.

Assume that we have collected L input and output data, and define the following cost function as

$$\begin{aligned} J(\boldsymbol{\theta}(L), \boldsymbol{\theta}(L-1), \dots, \boldsymbol{\theta}(1)) = {} & \sum_{k=1}^{L}[o(k) - \boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{\theta}(k)]^2 \\ & + \mu\sum_{k=1}^{L-1}[\boldsymbol{\theta}(k+1) - \boldsymbol{\theta}(k)]^{\mathrm{T}}[\boldsymbol{\theta}(k+1) - \boldsymbol{\theta}(k)]. \end{aligned} \tag{8}$$

Then, the smallest cost function at the sampling instant k is

$$\begin{aligned} \phi(\boldsymbol{\theta}(k+1); \mu, k) = {} & \inf_{\boldsymbol{\theta}(1),\dots,\boldsymbol{\theta}(k)} \left\{ \sum_{i=1}^{k}[o(i) - \boldsymbol{\varphi}^{\mathrm{T}}(i)\boldsymbol{\theta}(i)]^2 \right. \\ & \left. + \mu\sum_{i=1}^{k}[\boldsymbol{\theta}(i+1) - \boldsymbol{\theta}(i)]^{\mathrm{T}}[\boldsymbol{\theta}(i+1) - \boldsymbol{\theta}(i)] \right\} \\ = {} & \inf_{\boldsymbol{\theta}(k)} \left\{ [o(k) - \boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{\theta}(k)]^2 + \mu[\boldsymbol{\theta}(k+1) - \boldsymbol{\theta}(k)]^{\mathrm{T}}[\boldsymbol{\theta}(k+1) - \boldsymbol{\theta}(k)] + \phi(\boldsymbol{\theta}(k); \mu, k-1) \right\}, \end{aligned} \tag{9}$$

where the smallest cost function is written as a quadratic form

$$\phi(\boldsymbol{\theta}(k); \mu, k-1) = \boldsymbol{\theta}^{\mathrm{T}}(k)\mathbf{Q}(k-1)\boldsymbol{\theta}(k) - 2\boldsymbol{\theta}^{\mathrm{T}}(k)\mathbf{p}(k-1) + r(k-1), \tag{10}$$

in which $\mathbf{Q}(k-1) \in \mathbb{R}^{(m+n)\times(m+n)}$, $\mathbf{p}(k) \in \mathbb{R}^{m+n}$, and $r(k-1)$ is a scalar.

Substituting Equation (10) into Equation (9) and differentiating (9) with respect to $\boldsymbol{\theta}(k)$ yield,

$$\boldsymbol{\theta}(k) = \mathbf{e}(k) + \mathbf{M}(k)\boldsymbol{\theta}(k+1), \tag{11}$$

where

$$\begin{aligned} \mathbf{e}(k) &= \frac{1}{\mu}\mathbf{M}(k)(\mathbf{p}(k-1) + \boldsymbol{\varphi}(k)o(k)), \\ \mathbf{M}(k) &= \mu(\mathbf{Q}(k-1) + \mu\mathbf{I} + \boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k))^{-1}, \end{aligned}$$

the initial $\mathbf{Q}(0) = \mathbf{0} \in \mathbb{R}^{(m+n)\times(m+n)}$ and $\mathbf{p}(0) = \mathbf{0} \in \mathbb{R}^{m+n}$.

It follows that

$$\phi(\boldsymbol{\theta}(k+1); \mu, k) = \boldsymbol{\theta}^{\mathrm{T}}(k+1)\mathbf{Q}(k)\boldsymbol{\theta}(k+1) - 2\boldsymbol{\theta}^{\mathrm{T}}(k+1)\mathbf{p}(k) + r(k), \tag{12}$$

in which

$$\begin{aligned} \mathbf{Q}(k) &= \mu(\mathbf{I} - \mathbf{M}(k)), \\ \mathbf{p}(k) &= \mu\mathbf{e}(k), \\ r(k) &= r(k-1) + o^2(k) - (\mathbf{p}(k-1) + \boldsymbol{\varphi}(k)o(k))^{\mathrm{T}}\mathbf{e}(k). \end{aligned}$$

According to Equation (11), one should obtain $\boldsymbol{\theta}(k+1)$ first in order to compute $\boldsymbol{\theta}(k)$. The estimates $\boldsymbol{\theta}(L|L,\mu)$ can be obtained based on Equation (9),

$$\boldsymbol{\theta}(L|L,\mu) = (\mathbf{Q}(L-1) + \boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{\varphi}(k))^{-1}(\mathbf{p}(k-1) + \boldsymbol{\varphi}(k)o(k)),$$

where the first L in $\boldsymbol{\theta}(L|L,\mu)$ means the sampling instant L, and the second L means the estimates are conditional on the observations $o(1),\dots,o(L)$.

The filtered FLS estimates $\boldsymbol{\theta}^{\mathrm{FLS}}(1|\mu), \boldsymbol{\theta}^{\mathrm{FLS}}(2|\mu),\dots,\boldsymbol{\theta}^{\mathrm{FLS}}(L|\mu)$ can be sequentially obtained by

$$\boldsymbol{\theta}^{\mathrm{FLS}}(k|\mu) = \underset{\boldsymbol{\theta}(k)}{\operatorname{argmin}}[(o(k) - \boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{\theta}(k))^2 + \phi(\boldsymbol{\theta}(k);\mu,k-1)].$$

Since

$$\mathbf{e}(k) = [\mathbf{I} - \mathbf{M}(k)]\boldsymbol{\theta}^{\mathrm{FLS}}(k|\mu).$$

The FLS smoothed estimates for $\boldsymbol{\theta}(k)$ can be updated in reverse order based on Equation (11)

$$\begin{aligned}
\boldsymbol{\theta}(L-1|L,\mu) &= (\mathbf{I} - \mathbf{M}(k))\boldsymbol{\theta}^{\mathrm{FLS}}(L-1|\mu) + \mathbf{M}(k)\boldsymbol{\theta}(L|L,\mu),\\
\boldsymbol{\theta}(L-2|L,\mu) &= (\mathbf{I} - \mathbf{M}(k))\boldsymbol{\theta}^{\mathrm{FLS}}(L-2|\mu) + \mathbf{M}(k)\boldsymbol{\theta}(L-1|L,\mu),\\
&\vdots\\
\boldsymbol{\theta}(1|L,\mu) &= (\mathbf{I} - \mathbf{M}(k))\boldsymbol{\theta}^{\mathrm{FLS}}(1|\mu) + \mathbf{M}(k)\boldsymbol{\theta}(2|L,\mu).
\end{aligned}$$

The FLS method proposed by Kalaba and Tesfatsion can obtain the time varying parameter estimates, but it has the following shortcomings:

- Using the parameter estimates $\boldsymbol{\theta}(k+1|L,\mu)$ to obtain $\boldsymbol{\theta}(k|L,\mu)$ is unrealistic because the input-output data are collected in sequence when performing the on-line identification algorithms.
- The computational efforts are heavy, for example, when computing the parameter estimates $\boldsymbol{\theta}(k|L,\mu)$, one should compute $\boldsymbol{\theta}(k+1|L,\mu)$ first which involves matrix inversion, and computing $\mathbf{M}(k), \mathbf{Q}(k), \mathbf{p}(k)$ also leads to heavy computational costs.
- Estimate the parameters using a complex algorithm structure, for example, at sampling instant k, one should use the stored $\mathbf{Q}(k-1), \mathbf{p}(k-1)$ to get $\mathbf{Q}(k), \mathbf{M}(k), \mathbf{p}(k)$, and then obtain the parameter estimates $\boldsymbol{\theta}(k|L,\mu)$ based on $\mathbf{Q}(k), \mathbf{p}(k)$, $\boldsymbol{\theta}(k+1|L,\mu)$ and $\boldsymbol{\theta}(k|\mu)$.

4 | THE AUGMENTED FLS ALGORITHM

In this section, an augmented on-line FLS algorithm is proposed, which can estimate the parameters with less computational efforts and simple structures.

4.1 | Off-line FLS algorithm

Define

$$\begin{aligned}
\boldsymbol{\Phi}(L) &= [\boldsymbol{\varphi}(L),\dots,\boldsymbol{\varphi}(1)] \in \mathbb{R}^{(m+n)\times L},\\
\boldsymbol{\Theta}(L) &= [\boldsymbol{\theta}^{\mathrm{T}}(L),\dots,\boldsymbol{\theta}^{\mathrm{T}}(1)]^{\mathrm{T}} \in \mathbb{R}^{(Lm+Ln)\times 1},\\
\boldsymbol{\Psi}(L) &= \begin{bmatrix} \boldsymbol{\varphi}(L) & \cdots & \mathbf{0}\\ \vdots & \ddots & \vdots\\ \mathbf{0} & \cdots & \boldsymbol{\varphi}(1) \end{bmatrix} \in \mathbb{R}^{(Lm+Ln)\times L}.
\end{aligned}$$

The time-varying parameter system can be rewritten by

$$\mathbf{O}(L) = \boldsymbol{\Psi}^{\mathrm{T}}(L)\boldsymbol{\Theta}(L) + \mathbf{V}(L).$$

To get the parameter estimates for $\boldsymbol{\Theta}(L)$, let

$$\mathbf{R}_k(\mu) = \begin{cases} \boldsymbol{\varphi}(1)\boldsymbol{\varphi}^{\mathrm{T}}(1) + \mu\mathbf{I}, & \textit{if } k = 1, \\ \boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k) + 2\mu\mathbf{I}, & \textit{if } k \neq 1,\ L, \\ \boldsymbol{\varphi}(L)\boldsymbol{\varphi}^{\mathrm{T}}(L) + \mu\mathbf{I}, & \textit{if } k = L, \end{cases} \tag{13}$$

$$\mathbf{R}(L,\mu) = \begin{bmatrix} \mathbf{R}_L(\mu) & -\mu\mathbf{I} & \mathbf{0} & \cdots & \mathbf{0} \\ -\mu\mathbf{I} & \mathbf{R}_{L-1}(\mu) & -\mu\mathbf{I} & \cdots & \mathbf{0} \\ \mathbf{0} & -\mu\mathbf{I} & \mathbf{R}_{L-2}(\mu) & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & -\mu\mathbf{I} \\ \mathbf{0} & \mathbf{0} & \cdots & -\mu\mathbf{I} & \mathbf{R}_1(\mu) \end{bmatrix} \in \mathbb{R}^{P\times P},$$

$$P = Lm + Ln. \tag{14}$$

Then, the cost function (8) is transformed into

$$J(\boldsymbol{\Theta}(L)) = \boldsymbol{\Theta}^{\mathrm{T}}(L)\mathbf{R}(L,\mu)\boldsymbol{\Theta}(L) - 2\boldsymbol{\Theta}^{\mathrm{T}}(L)\boldsymbol{\Psi}(L)\mathbf{O}(L) + \mathbf{O}^{\mathrm{T}}(L)\mathbf{O}(L).$$

Differentiating the cost function with respect to $\boldsymbol{\Theta}(L)$ yields,

$$\frac{\partial J(\boldsymbol{\Theta}(L))}{\boldsymbol{\Theta}(L)} = 2\mathbf{R}(L,\mu)\boldsymbol{\Theta}(L) - 2\boldsymbol{\Psi}(L)\mathbf{O}(L),$$

it shows that

$$\boldsymbol{\Theta}(L) = \mathbf{R}^{-1}(L,\mu)\boldsymbol{\Psi}(L)\mathbf{O}(L). \tag{15}$$

Remark 6. When $\mu > 0$ and $L \geqslant 1$, the $P \times P$ matrix $\mathbf{R}(L,\mu)$ is positive definite, thus the parameter estimates $\boldsymbol{\Theta}(L)$ can always be computed by Equation (15).

Remark 7. Although Equation (15) can get the parameter estimates for $\boldsymbol{\Theta}(L)$ in theory, it also leads to heavy computational efforts caused by the matrix inversion. When $\mathbf{R}(L,\mu) \in \mathbb{R}^{P\times P}$, the flops for the matrix inverse are $\mathbf{O}(P^3)$, for example, $P = 100$, the flops are 10^6.

The order of the matrix $\mathbf{R}(L,\mu)$ increases significantly with the increased number of the collected data. Therefore, using the off-line FLS algorithm for the time-varying parameter system is costly.

4.2 | On-line FLS algorithm

Unlike the off-line algorithm, the on-line algorithm can update the parameters based on the new arrived data with less computational efforts. The on-line algorithm is sensitive to the new data, thus, the varying feature of the system can be easily observed by the on-line algorithm.

Assume that the input-output data $u(k)$ and $o(k)$ have been collected at the sampling instant k, and the parameter estimates $\theta^{\mathrm{FLS}}(k-1)$ at $k-1$ have been obtained. One aims to get the parameter estimates $\theta^{\mathrm{FLS}}(k)$ based on the new collected data and the estimated parameters.

Define the cost function at the sampling instant k as

$$J(\boldsymbol{\theta}(k)) = \sum_{i=1}^{k}[o(i) - \boldsymbol{\varphi}^{\mathrm{T}}(i)\boldsymbol{\theta}(i)]^2 + \mu\sum_{i=1}^{k-1}[\boldsymbol{\theta}(i+1) - \boldsymbol{\theta}(i)]^{\mathrm{T}}[\boldsymbol{\theta}(i+1) - \boldsymbol{\theta}(i)], \tag{16}$$

which can be transformed into

$$\begin{aligned} J(\boldsymbol{\theta}(k)) = {} & [o(k) - \boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{\theta}(k)]^2 + \sum_{i=1}^{k-1}[o(i) - \boldsymbol{\varphi}^{\mathrm{T}}(i)\boldsymbol{\theta}(i)]^2 + \mu[\boldsymbol{\theta}(k) - \boldsymbol{\theta}(k-1)]^{\mathrm{T}} \\ & \times [\boldsymbol{\theta}(k) - \boldsymbol{\theta}(k-1)] + \mu\sum_{i=1}^{k-2}[\boldsymbol{\theta}(i+1) - \boldsymbol{\theta}(i)]^{\mathrm{T}}[\boldsymbol{\theta}(i+1) - \boldsymbol{\theta}(i)]. \end{aligned} \tag{17}$$

Since all the parameter estimates before k have been obtained, the above cost function is equivalent to

$$J(\boldsymbol{\theta}(k)) \propto [o(k) - \boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{\theta}(k)]^2 + \mu[\boldsymbol{\theta}(k) - \boldsymbol{\theta}^{\mathrm{FLS}}(k-1)]^{\mathrm{T}}[\boldsymbol{\theta}(k) - \boldsymbol{\theta}^{\mathrm{FLS}}(k-1)]. \tag{18}$$

Differentiating $J(\boldsymbol{\theta}(k))$ with respect to $\boldsymbol{\theta}(k)$ and letting it equal to zero yield,

$$\boldsymbol{\theta}^{\mathrm{FLS}}(k) = [\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k) + \mu\mathbf{I}]^{-1}[\boldsymbol{\varphi}(k)o(k) + \mu\boldsymbol{\theta}^{\mathrm{FLS}}(k-1)].$$

To derive the basic recurrence relation between $\boldsymbol{\theta}^{\mathrm{FLS}}(k)$ and $\boldsymbol{\theta}^{\mathrm{FLS}}(k-1)$, the above equation is written by

$$\begin{aligned}\boldsymbol{\theta}^{\mathrm{FLS}}(k) &= [\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k) + \mu\mathbf{I}]^{-1}[\boldsymbol{\varphi}(k)o(k) + (\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k) + \mu\mathbf{I})\boldsymbol{\theta}^{\mathrm{FLS}}(k-1) \\ &\quad - \boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{\theta}^{\mathrm{FLS}}(k-1)] \\ &= \boldsymbol{\theta}^{\mathrm{FLS}}(k-1) + [\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k) + \mu\mathbf{I}]^{-1}\boldsymbol{\varphi}(k)[o(k) - \boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{\theta}^{\mathrm{FLS}}(k-1)].\end{aligned} \tag{19}$$

Therefore, the on-line augmented FLS algorithm can be summarized as follows

$$\boldsymbol{\theta}^{\mathrm{FLS}}(k) = \boldsymbol{\theta}^{\mathrm{FLS}}(k-1) + \mathbf{M}^{-1}(k)\boldsymbol{\varphi}(k)e(k), \tag{20}$$

$$\mathbf{M}(k) = \boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k) + \mu\mathbf{I}, \tag{21}$$

$$e(k) = o(k) - \boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{\theta}^{\mathrm{FLS}}(k-1). \tag{22}$$

The on-line augmented FLS algorithm consists of the following steps.

1. Assign $u(k) = o(k) = 0$, $k \leqslant 0$, $\boldsymbol{\theta}^{\mathrm{FLS}}(0) = \mathbf{1}/10^6$ with $\mathbf{1} = [1, 1, \dots, 1]^{\mathrm{T}} \in \mathbb{R}^{m+n}$.
2. Let $k = 1$.
3. Generate the input-output data $u(k)$ and $o(k)$.
4. Form $\boldsymbol{\varphi}(k)$ by Equation (6).
5. Compute $e(k)$ by Equation (22).
6. Form $\mathbf{M}(k)$ by Equation (21) and compute its inverse matrix.
7. Update the parameter estimates $\boldsymbol{\theta}^{\mathrm{FLS}}(k)$ according to Equation (20).
8. Let $k = k + 1$ and go to step 3.

Remark 8. When $\mu > 0$, the matrix $[\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k) + \mu\mathbf{I}]$ is positive definite, thus the parameter estimates $\boldsymbol{\theta}^{\mathrm{FLS}}(k)$ can always be computed by Equation (20).

Remark 9. The order of the matrix $[\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k) + \mu\mathbf{I}]$ is $m + n$, which is much smaller than $L(m + n)$. Therefore, the computational efforts of the on-line FLS algorithm are less heavy than those of the off-line FLS algorithm.

4.3 | The smoothed on-line FLS algorithm

Assume that the initial parameter estimates $\overline{\boldsymbol{\theta}}(1), \dots, \overline{\boldsymbol{\theta}}(L)$ are known in prior. Then, the cost function is written by

$$J(\boldsymbol{\theta}(k)) = \sum_{i=1}^{k}[o(i) - \boldsymbol{\varphi}^{\mathrm{T}}(i)\boldsymbol{\theta}(i)]^2 + \mu\sum_{i=1}^{k}[\boldsymbol{\theta}(i+1) - \boldsymbol{\theta}(i)]^{\mathrm{T}}[\boldsymbol{\theta}(i+1) - \boldsymbol{\theta}(i)], \tag{23}$$

and can be simplified as follows

$$\begin{aligned}J(\boldsymbol{\theta}(k)) &\propto [o(k) - \boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{\theta}(k)]^2 + \mu[\overline{\boldsymbol{\theta}}(k+1) - \boldsymbol{\theta}(k)]^{\mathrm{T}}[\overline{\boldsymbol{\theta}}(k+1) \\ &\quad - \boldsymbol{\theta}(k)] + \mu[\boldsymbol{\theta}(k) - \boldsymbol{\theta}^{\mathrm{FLS}}(k-1)]^{\mathrm{T}}[\boldsymbol{\theta}(k) - \boldsymbol{\theta}^{\mathrm{FLS}}(k-1)].\end{aligned} \tag{24}$$

It follows that the smoothed parameter estimates $\boldsymbol{\theta}^{\mathrm{FLS}}(k)$ are computed by

$$\boldsymbol{\theta}^{\mathrm{FLS}}(k) = [\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k) + 2\mu\mathbf{I}]^{-1}[\boldsymbol{\varphi}(k)o(k) + \mu\boldsymbol{\theta}^{\mathrm{FLS}}(k-1) + \mu\overline{\boldsymbol{\theta}}(k+1)], \tag{25}$$

which can be transformed into

$$\begin{aligned}\boldsymbol{\theta}^{\mathrm{FLS}}(k) &= \frac{1}{2}[\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k)+2\mu\mathbf{I}]^{-1}[2\boldsymbol{\varphi}(k)o(k)+2\mu\boldsymbol{\theta}^{\mathrm{FLS}}(k-1)+2\mu\overline{\boldsymbol{\theta}}(k+1)]\\ &= \frac{1}{2}\boldsymbol{\theta}^{\mathrm{FLS}}(k-1)+\frac{1}{2}\overline{\boldsymbol{\theta}}(k+1)+\frac{1}{2}[\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k)+2\mu\mathbf{I}]^{-1}\boldsymbol{\varphi}(k)[o(k)\\ &\quad-\boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{\theta}^{\mathrm{FLS}}(k-1)]+\frac{1}{2}[\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k)+2\mu\mathbf{I}]^{-1}\boldsymbol{\varphi}(k)[o(k)-\boldsymbol{\varphi}^{\mathrm{T}}(k)\overline{\boldsymbol{\theta}}(k+1)].\end{aligned} \tag{26}$$

Remark 10. Equation (26) shows that the estimates $\boldsymbol{\theta}^{\mathrm{FLS}}(k-1)$ and $\overline{\boldsymbol{\theta}}(k+1)$ play the same role in estimating $\boldsymbol{\theta}^{\mathrm{FLS}}(k)$. However, when the data are collected in sequence, there is no quantifiable confidence of $\overline{\boldsymbol{\theta}}(k+1)$ when updating the parameters $\boldsymbol{\theta}(k)$. We usually use Equation (20) to compute the parameter estimates $\boldsymbol{\theta}^{\mathrm{FLS}}(k)$, or assign a smaller weight for the estimates $\overline{\boldsymbol{\theta}}(k+1)$, for example,

$$\begin{aligned}\boldsymbol{\theta}^{\mathrm{FLS}}(k) &= \lambda\boldsymbol{\theta}^{\mathrm{FLS}}(k-1)+(1-\lambda)\overline{\boldsymbol{\theta}}(k+1)+\lambda[\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k)+2\mu\mathbf{I}]^{-1}\boldsymbol{\varphi}(k)\\ &\quad\times[o(k)-\boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{\theta}^{\mathrm{FLS}}(k-1)]+(1-\lambda)[\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k)+2\mu\mathbf{I}]^{-1}\boldsymbol{\varphi}(k)[o(k)-\boldsymbol{\varphi}^{\mathrm{T}}(k)\overline{\boldsymbol{\theta}}(k+1)],\\ &\quad\frac{1}{2}<\lambda<1.\end{aligned} \tag{27}$$

Assume that L input–output data have been collected. Let the initial parameter estimates be $\overline{\boldsymbol{\theta}}_0(1),\dots,\overline{\boldsymbol{\theta}}_0(L)$. Based on Equation (27), the smoothed FLS estimates in the first iteration can be computed by

$$\begin{aligned}\boldsymbol{\theta}_1^{\mathrm{FLS}}(k) &= \lambda\boldsymbol{\theta}_1^{\mathrm{FLS}}(k-1)+(1-\lambda)\overline{\boldsymbol{\theta}}_0(k+1)+\lambda[\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k)+2\mu\mathbf{I}]^{-1}\boldsymbol{\varphi}(k)\\ &\quad\times[o(k)-\boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{\theta}_1^{\mathrm{FLS}}(k-1)]+(1-\lambda)[\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k)+2\mu\mathbf{I}]^{-1}\boldsymbol{\varphi}(k)[o(k)-\boldsymbol{\varphi}^{\mathrm{T}}(k)\overline{\boldsymbol{\theta}}_0(k+1)],\\ &\quad k=1,\dots,L-1,\ \frac{1}{2}<\lambda<1,\end{aligned} \tag{28}$$

and

$$\begin{aligned}\boldsymbol{\theta}_1^{\mathrm{FLS}}(L) &= \lambda\boldsymbol{\theta}_1^{\mathrm{FLS}}(L-1)+\lambda[\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k)+2\mu\mathbf{I}]^{-1}\boldsymbol{\varphi}(k)[o(L)-\boldsymbol{\varphi}^{\mathrm{T}}(L)\boldsymbol{\theta}_1^{\mathrm{FLS}}(L-1)]\\ &\quad+(1-\lambda)\overline{\boldsymbol{\theta}}_0(L),\quad \frac{1}{2}<\lambda<1.\end{aligned} \tag{29}$$

Once all the parameter estimates $\boldsymbol{\theta}_1^{\mathrm{FLS}}(k), k=1,\dots,L$ have been estimated, let

$$\overline{\boldsymbol{\theta}}_1(k)=\boldsymbol{\theta}_1^{\mathrm{FLS}}(k),\quad k=1,\dots,L.$$

In the second iteration, we have more confidence of $\overline{\boldsymbol{\theta}}_1(k), k=1,\dots,L$ than $\overline{\boldsymbol{\theta}}_0(k)$ in the first iteration. Thus, $\boldsymbol{\theta}_2^{\mathrm{FLS}}(k)$ can be computed by

$$\begin{aligned}\boldsymbol{\theta}_2^{\mathrm{FLS}}(k) &= \lambda^2\boldsymbol{\theta}_2^{\mathrm{FLS}}(k-1)+(1-\lambda^2)\overline{\boldsymbol{\theta}}_1(k+1)+\lambda^2[\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k)+2\mu\mathbf{I}]^{-1}\boldsymbol{\varphi}(k)\\ &\quad\times[o(k)-\boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{\theta}_2^{\mathrm{FLS}}(k-1)]+(1-\lambda^2)[\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k)+2\mu\mathbf{I}]^{-1}\boldsymbol{\varphi}(k)[o(k)-\boldsymbol{\varphi}^{\mathrm{T}}(k)\overline{\boldsymbol{\theta}}_1(k+1)],\\ &\quad\frac{1}{2}<\lambda<1,\ k=1,\dots,L-1,\\ \boldsymbol{\theta}_2^{\mathrm{FLS}}(L) &= \lambda^2\boldsymbol{\theta}_2^{\mathrm{FLS}}(L-1)+\lambda^2[\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k)+2\mu\mathbf{I}]^{-1}\boldsymbol{\varphi}(k)\\ &\quad\times[(o(L)-\boldsymbol{\varphi}^{\mathrm{T}}(L)\boldsymbol{\theta}_2^{\mathrm{FLS}}(L-1)]+(1-\lambda^2)\overline{\boldsymbol{\theta}}_1(L).\end{aligned} \tag{30}$$

In the following iterations, the estimates $\overline{\boldsymbol{\theta}}_{l-1}(k), k=1,\dots,L, l\geqslant 3$ are more and more accurate. To get the FLS estimates, a larger weight should be assigned to the estimates $\overline{\boldsymbol{\theta}}_{l-1}(k)$ when computing $\boldsymbol{\theta}_l^{\mathrm{FLS}}(k-1)$, for example, λ^l becomes smaller and smaller with the increase of the iteration l. In general, the augmented smoothed FLS estimates are written by

$$\begin{aligned}\boldsymbol{\theta}_l^{\mathrm{FLS}}(k) &= \lambda^l\boldsymbol{\theta}_l^{\mathrm{FLS}}(k-1)+(1-\lambda^l)\overline{\boldsymbol{\theta}}_{l-1}(k+1)+\lambda^l[\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k)+2\mu\mathbf{I}]^{-1}\boldsymbol{\varphi}(k)\\ &\quad\times[o(k)-\boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{\theta}_l^{\mathrm{FLS}}(k-1)]+(1-\lambda^l)[\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k)+2\mu\mathbf{I}]^{-1}\boldsymbol{\varphi}(k)[o(k)-\boldsymbol{\varphi}^{\mathrm{T}}(k)\overline{\boldsymbol{\theta}}_{l-1}(k+1)],\end{aligned}$$

$$k = 1, \ldots, L-1, \frac{1}{2} < \lambda < 1,$$

$$\boldsymbol{\theta}_l^{\mathrm{FLS}}(L) = \lambda^l \boldsymbol{\theta}_l^{\mathrm{FLS}}(L-1) + \lambda^l [\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k) + 2\mu \mathbf{I}]^{-1} \boldsymbol{\varphi}(k)[o(L) - \boldsymbol{\varphi}^{\mathrm{T}}(L)\boldsymbol{\theta}^{\mathrm{FLS}}(L-1)]$$
$$+ (1-\lambda^l)\overline{\boldsymbol{\theta}}_{l-1}(L).$$

The steps of the augmented smoothed FLS algorithm are listed as follows.

1. Let $u(i) = 0, o(i) = 0, i \leqslant 0, \boldsymbol{\theta}^{\mathrm{FLS}}(0) = \mathbf{1}/10^6$ and $\overline{\boldsymbol{\theta}}_0(k) = \mathbf{1}/10^6, k = 1, \ldots, L$ with $\mathbf{1} = [1, 1, \ldots, 1]^{\mathrm{T}} \in \mathbb{R}^{m+n}$.
2. Assign a positive constant λ, $0.5 < \lambda < 1$, a positive constant μ and a small positive number ε.
3. Generate $u(1), \ldots, u(L), o(1), \ldots, o(L)$, let $l = 1$.
4. Let $k = 1$, and compute λ^l, if $\lambda^l < \varepsilon$, then go to step 10; Otherwise, go to next step.
5. Form $\boldsymbol{\varphi}(k)$ and compute $[\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k) + 2\mu\mathbf{I}]^{-1}$.
6. Compute $o(k) - \boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{\theta}_l^{\mathrm{FLS}}(k-1)$.
7. Compute $o(k) - \boldsymbol{\varphi}^{\mathrm{T}}(k)\overline{\boldsymbol{\theta}}_{l-1}(k+1)$.
8. Update $\boldsymbol{\theta}_l^{\mathrm{FLS}}(k)$.
9. Increase k by 1, if $k \leqslant L$, go to step 5; Otherwise, increase l by 1 and let $\overline{\boldsymbol{\theta}}_{l-1}(k) = \boldsymbol{\theta}_{l-1}^{\mathrm{FLS}}(k), k = 1, \ldots, L$, then go to step 4.
10. Obtain the smoothed FLS estimates $\boldsymbol{\theta}_l^{\mathrm{FLS}}(k) = \overline{\boldsymbol{\theta}}_{l-1}(k), k = 1, \ldots, L$.

Remark 11. In iteration l, the smoothed estimates $\boldsymbol{\theta}_l^{\mathrm{FLS}}(k)$ are calculated based on $\overline{\boldsymbol{\theta}}_{l-1}(k+1)$ and $\boldsymbol{\theta}_l^{\mathrm{FLS}}(k-1)$ in proper order. Although k continually changes as new measurements are collected, the lag $k+1-k=1$ is a constant. That is, at each sampling instant k, we have 1 future measurement $\overline{\boldsymbol{\theta}}_{l-1}(k+1)$ available. This type of smoothing method is usually termed as fixed-lag smoothing method.[35]

5 | PROPERTIES OF THE AUGMENTED FLS ALGORITHM

The augmented FLS estimates reflect the prior belief that the parameter vectors $\boldsymbol{\theta}(k)$, evolve only slowly over time, if at all. What would happen if the parameter vector performs a single unanticipated shift at time k? and what is the difference between the Kalman filter and the augmented FLS algorithm? In this section, some properties of the augmented FLS algorithm are given.

5.1 | The relation between the FLS algorithm and the Kalman filter

Since the parameter vectors evolve slowly over time, the time-varying parameter systems are written by

State model:

$$\boldsymbol{\theta}(k) = \boldsymbol{\theta}(k-1) + \mathbf{w}(k). \tag{31}$$

Output model:

$$o(k) = \boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{\theta}(k) + v(k). \tag{32}$$

Assume that $\mathbf{w}(k)$ and $v(k)$ are independent Gaussian white noises with variances $\sigma^2\mathbf{I}$ and ς^2, respectively.

Let $\overline{\boldsymbol{\theta}}(k)$ be the estimates conditioned on the input data $u(1), \ldots, u(k)$, the output data $o(1), \ldots, o(k-1)$ and the filtered estimates $\boldsymbol{\theta}^{\mathrm{FLS}}(k-1)$. According to the state model, we have

$$\overline{\boldsymbol{\theta}}(k) = \boldsymbol{\theta}^{\mathrm{FLS}}(k-1). \tag{33}$$

When the output $o(k)$ is collected, the filtered estimates $\boldsymbol{\theta}^{\mathrm{FLS}}(k)$ are computed by

$$\boldsymbol{\theta}^{\mathrm{FLS}}(k) = \overline{\boldsymbol{\theta}}(k) + L_k[o(k) - \boldsymbol{\varphi}^{\mathrm{T}}(k)\overline{\boldsymbol{\theta}}(k)] = \boldsymbol{\theta}^{\mathrm{FLS}}(k-1) + L_k[o(k) - \boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{\theta}^{\mathrm{FLS}}(k-1)], \tag{34}$$

where

$$\mathbf{L}_k = \mathbf{P}_k^- \boldsymbol{\varphi}(k)[\boldsymbol{\varphi}^{\mathrm{T}}(k)\mathbf{P}_k^- \boldsymbol{\varphi}(k) + \varsigma^2]^{-1},$$
$$\mathbf{P}_k^- = \mathbf{P}_{k-1} + \sigma^2\mathbf{I},$$
$$\mathbf{P}_{k-1} = (\mathbf{I} - \mathbf{L}_{k-1}\boldsymbol{\varphi}^{\mathrm{T}}(k))\mathbf{P}_{k-1}^-.$$

Rewrite the FLS estimates $\boldsymbol{\theta}^{\mathrm{FLS}}(k)$ as follows,

$$\boldsymbol{\theta}_{\mathrm{FLS}}(k) = \boldsymbol{\theta}^{\mathrm{FLS}}(k-1) + [\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k) + \mu\mathbf{I}]^{-1}\boldsymbol{\varphi}(k)(o(k) - \boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{\theta}^{\mathrm{FLS}}(k-1)). \tag{35}$$

Clearly, we can conclude that the FLS estimates have the same structure with the filtered estimates, and if

$$[\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k) + \mu\mathbf{I}]^{-1} = \mathbf{P}_k^-[\boldsymbol{\varphi}^{\mathrm{T}}(k)\mathbf{P}_k^-\boldsymbol{\varphi}(k) + \varsigma^2]^{-1},$$

the FLS estimates are equal to the filtered estimates.

Remark 12. The Kalman filter obtains a unique estimate for the state sequence based on probability assumptions for model discrepancy terms; while the FLS algorithm yields a family of state sequence estimates, and each of which is vector-minimally incompatible with the prior dynamical and measurement specifications.[21]

5.2 | The weight μ in the FLS algorithm

In this subsection, the role of the weight μ in the FLS algorithm will be considered.

1. When $\mu = 0$, the cost function is written by

$$J(\boldsymbol{\theta}(L), \boldsymbol{\theta}(L-1), \dots, \boldsymbol{\theta}(1)) = \sum_{k=1}^{L}[o(k) - \boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{\theta}(k)]^2, \tag{36}$$

and the FLS estimates are

$$\boldsymbol{\theta}^{\mathrm{FLS}}(k) = \boldsymbol{\theta}^{\mathrm{FLS}}(k-1) + [\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k)]^{-1}\boldsymbol{\varphi}(k)(o(k) - \boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{\theta}^{\mathrm{FLS}}(k-1)). \tag{37}$$

Both Equations (36) and (37) demonstrate that the FLS algorithm is unavailable for time-varying parameter systems when $\mu = 0$: the derivative function of Equation (36) has no unique solution, and $\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k)$ in Equation (37) is a singular matrix.

Remark 13. $\mu = 0$ indicates that the parameters of the systems shift intensively with high frequency, where the FLS algorithm is inefficient for such systems.

2. When $\mu \to \infty$, the cost function is defined as

$$J(\boldsymbol{\theta}(L), \boldsymbol{\theta}(L-1), \dots, \boldsymbol{\theta}(1)) = \sum_{k=1}^{L-1}[\boldsymbol{\theta}(k+1) - \boldsymbol{\theta}(k)]^{\mathrm{T}}[\boldsymbol{\theta}(k+1) - \boldsymbol{\theta}(k)], \tag{38}$$

and the FLS estimates are computed by

$$\boldsymbol{\theta}^{\mathrm{FLS}}(k) = \boldsymbol{\theta}^{\mathrm{FLS}}(k-1), \tag{39}$$

which means that the parameter estimates keep unchanged, for example, $\boldsymbol{\theta}^{\mathrm{FLS}}(k) = \boldsymbol{\theta}^{\mathrm{FLS}}(0), k = 1, 2, \dots, L$.

Remark 14. If the parameters of the systems evolve slowly, a larger μ will be assigned to $\sum_{k=1}^{L-1}[\boldsymbol{\theta}(k+1) - \boldsymbol{\theta}(k)]^{\mathrm{T}}[\boldsymbol{\theta}(k+1) - \boldsymbol{\theta}(k)]$; Otherwise, a small μ is better.

5.3 | The FLS algorithm for the switching systems

Suppose that the switching system is defined as

$$o(i) = \boldsymbol{\varphi}(i)\boldsymbol{\theta}_a, \quad i = 1, \dots, k-1,$$
$$o(j) = \boldsymbol{\varphi}(j)\boldsymbol{\theta}_b, \quad j = k, \dots, L,$$

where k is the switching instant, and L is the number of the total collected data. $\boldsymbol{\theta}_a$ and $\boldsymbol{\theta}_b$ are the true parameters of the two sub-models, respectively.

Rewrite the cost function

$$J(\boldsymbol{\theta}(k)) = [o(k) - \boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{\theta}(k)]^2 + \mu[\boldsymbol{\theta}(k) - \boldsymbol{\theta}^{\mathrm{FLS}}(k-1)]^{\mathrm{T}}[\boldsymbol{\theta}(k) - \boldsymbol{\theta}^{\mathrm{FLS}}(k-1)].$$

Clearly, if k is large enough, both the two terms on the right side of the above equation can keep the estimates $\boldsymbol{\theta}^{\mathrm{FLS}}(k-1)$ converging to $\boldsymbol{\theta}_a$. However, at the sampling instant k, the first term on the right side of the above equation will play a major role first (its error is far larger than that of the later one), which will force the estimates to diverge from $\boldsymbol{\theta}_a$, and then the two terms will keep the estimates $\boldsymbol{\theta}(j)$ slowly converging to the true values $\boldsymbol{\theta}_b$ if $L \gg k$.

Remark 15. For a switching system whose parameters perform a shift intensively at some sampling instants k, the FLS algorithm is also effective when the numbers ($k-1$ and $L-k+1$) of the collected data for each sub-model are large enough.

Remark 16. The proposed algorithm in this article can combine other identification methods[36,37] to study new parameter estimation approaches for linear and nonlinear stochastic systems,[38,39] and can be applied to other literatures,[40,41] such as chemical process control systems and paper-making systems.

6 | SIMULATION EXAMPLES

6.1 | Example 1

Consider the time-varying parameter model in Reference 1,

$$o(k) = b_1(k)u_1(k) + b_2(k)u_2(k) + v(k),$$
$$b_1(k) = \sin\left(\frac{2\pi}{15}k\right), \quad b_2(k) = \cos\left(\frac{\pi}{15}k\right),$$
$$u_1(k) = \sin(10+k) + 0.01, \quad u_2(k) = \cos(10+k).$$

Monte Carlo simulations (with 100 different noise seeds) are performed based on the on-line augmented FLS algorithm ($\mu = 1$, $L = 30$, $v(k) \backsim N(0, 0.01)$). The parameter estimates with their true values are shown in Figure 1. The output estimates, the true outputs, and their estimation errors are depicted in Figure 2. Clearly, the parameter estimates and output estimates can catch their true values whatever the noise is. This example shows that the on-line augmented FLS algorithm is robust to noise.

In addition, thanks to the second term in the cost function (16), the number of the collected data does not need to be larger than the number of the unknown parameters when using the on-line augmented FLS algorithm, which is different from the traditional LS algorithm. In this example, there exist two unknown parameters in each sampling instant k, then the total number of unknown parameters for $L = 30$ instants is 60, while the number of the collected data is 30. That is, the on-line augmented FLS algorithm can work for this time-varying model, but the traditional LS algorithm cannot.

6.2 | Rational model

Consider the rational model proposed in Reference 42,

$$o(k) = \frac{0.3o(k-1) + 0.7u(k-1)}{1 + u^2(k-1)} + v(k), \tag{40}$$

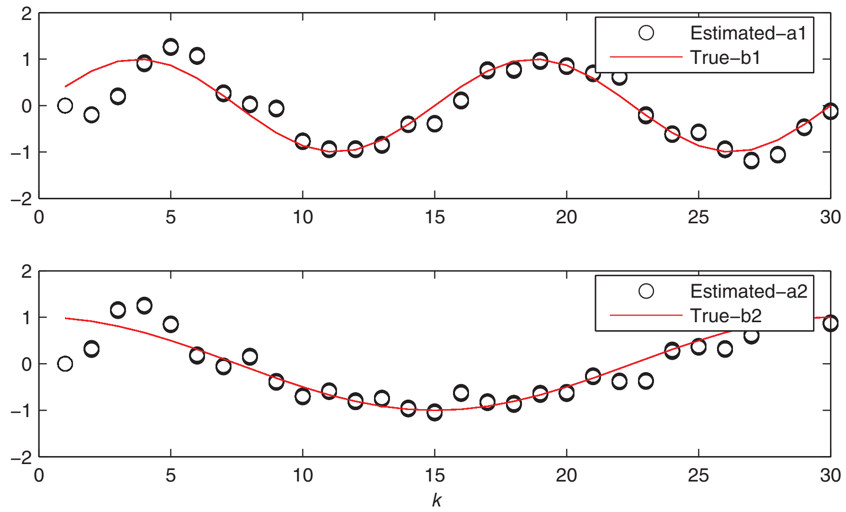

FIGURE 1 The true parameters and their estimates

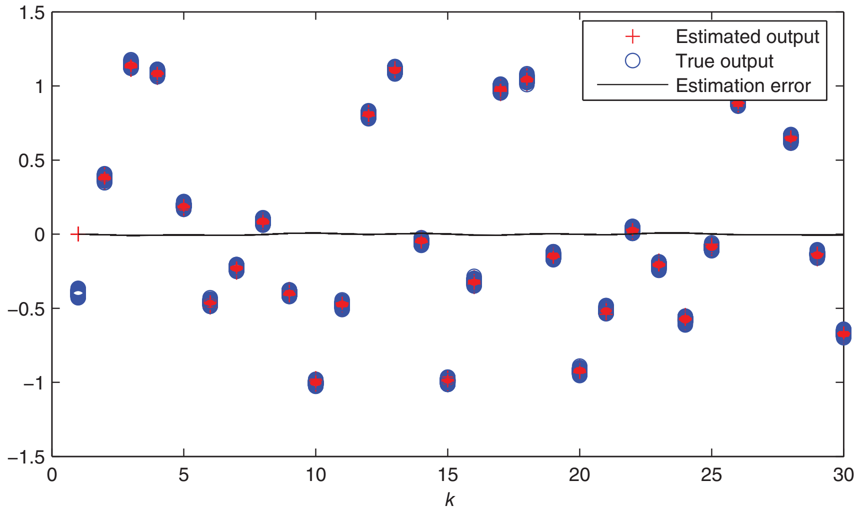

FIGURE 2 The output estimates, the true outputs, and their errors

where $v(k) \backsim N(0, 0.04)$. Assume that

$$b_1(k) = \frac{0.3}{1 + u^2(k-1)}, \quad b_2(k) = \frac{0.7}{1 + u^2(k-1)}.$$

In simulation, the input data are generated by $u(k) = \sin(\frac{\pi}{30}k)$. Use the traditional FLS (T-FLS) algorithm in Reference 1 and the on-line augmented FLS (A-FLS) algorithm proposed in this article for the rational model ($\mu = 0.5, L = 50$). The parameter estimates with their true values are shown in Figures 3 and 4. The output estimates, the true outputs, and their estimation errors are depicted in Figures 5 and 6. Since we use a smoothed term to adjust the estimates, see the third term in (24), the A-FLS algorithm proposed in this article has more accurate estimation accuracy than that of the T-FLS algorithm.

6.3 | Switching model

The switching model is written by

$$o(k) = 0.3o(k-1) + 0.7u(k-1) + v(k), \quad k = 1, 2, \ldots, 100,$$
$$o(k) = 0.8o(k-1) + u(k-1) + v(k), \quad k = 101, 102, \ldots, 200,$$

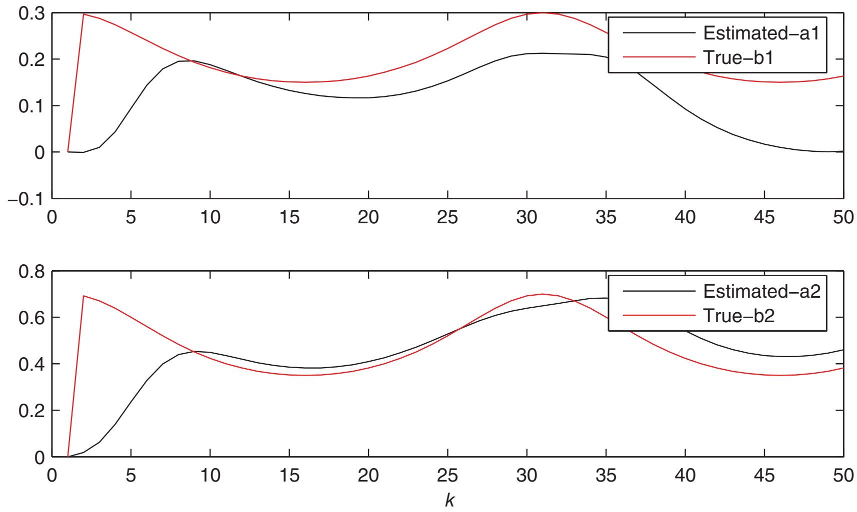

FIGURE 3 The true parameters and their estimates using T-FLS (rational model)

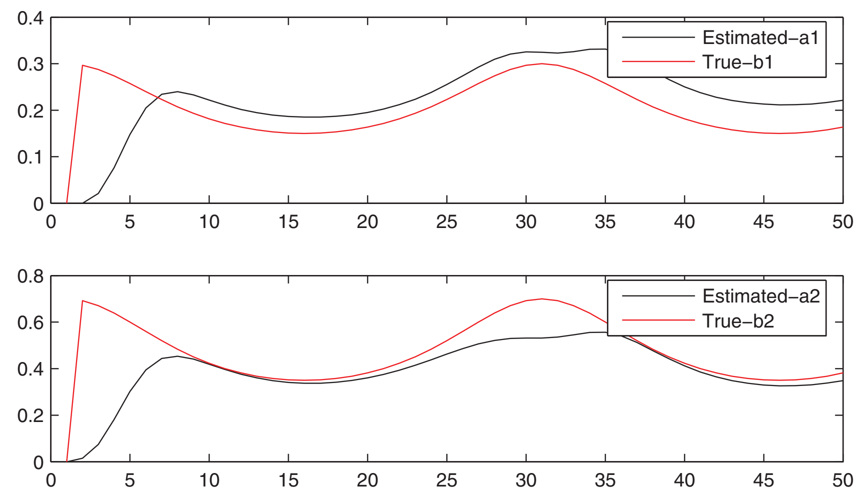

FIGURE 4 The true parameters and their estimates using A-FLS (rational model)

and can be simplified to a time-varying parameter system,

$$\begin{aligned} o(k) &= b_1(k)o(k-1) + b_2(k)u(k-1) + v(k), \\ b_1(k) &= 0.3, \quad b_2(k) = 0.7, \quad k = 1, 2, \dots, 100, \\ b_1(k) &= 0.8, \quad b_2(k) = 1, \quad k = 101, 102, \dots, 200, \end{aligned}$$

the input $\{u(k)\}$ satisfies $u(k) \sim N(0, 1)$, and the noise $\{v(k)\}$ satisfies $v(k) \sim N(0, 0.04)$.

Apply the on-line augmented FLS algorithm to the switching model ($\mu = 0.1$). The parameter estimates and the output estimates are shown in Figures 7 and 8, respectively. This example shows that the on-line augmented FLS algorithm is effective for switching models.

6.4 | ExpAR model

Consider the following ExpAR model proposed in Reference 43,

$$o(k) = (1.95 + 0.23e^{-o^2(k-1)})o(k-1) - (0.96 + 0.24e^{-o^2(k-1)})o(k-2) + v(k).$$

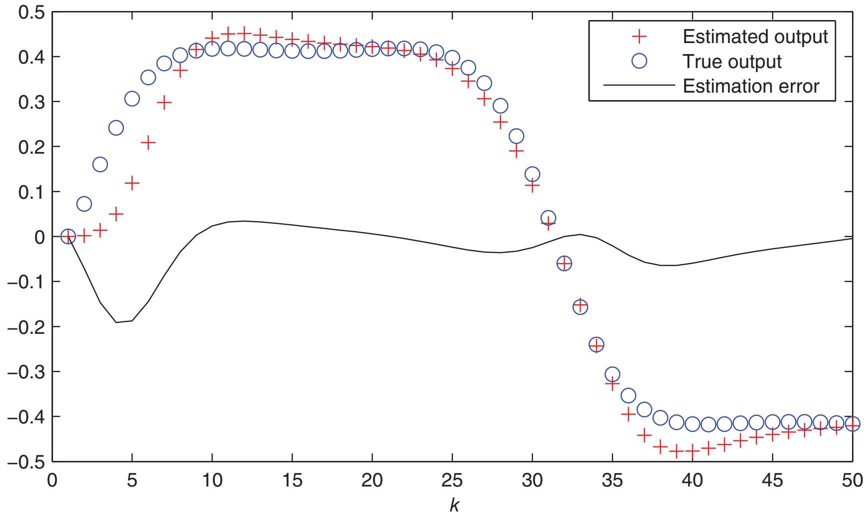

FIGURE 5 The output estimates, the true outputs, and their errors using T-FLS (rational model)

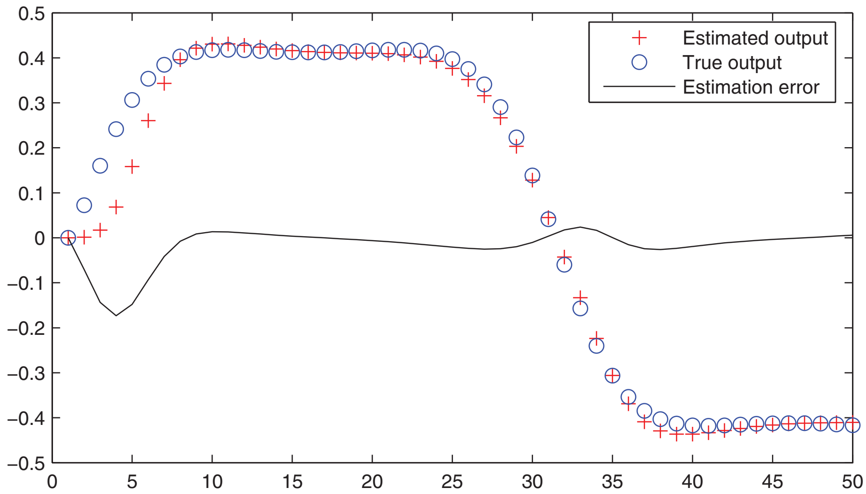

FIGURE 6 The output estimates, the true outputs, and their errors A-FLS (rational model)

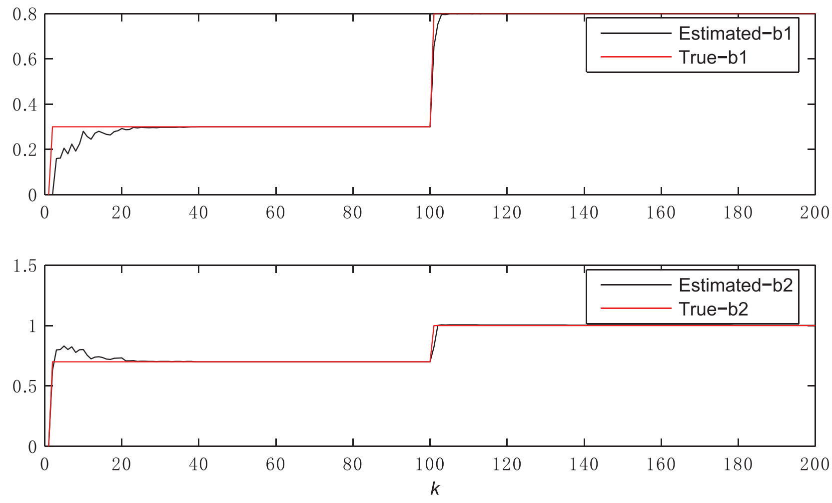

FIGURE 7 The true parameters and their estimates (switching model)

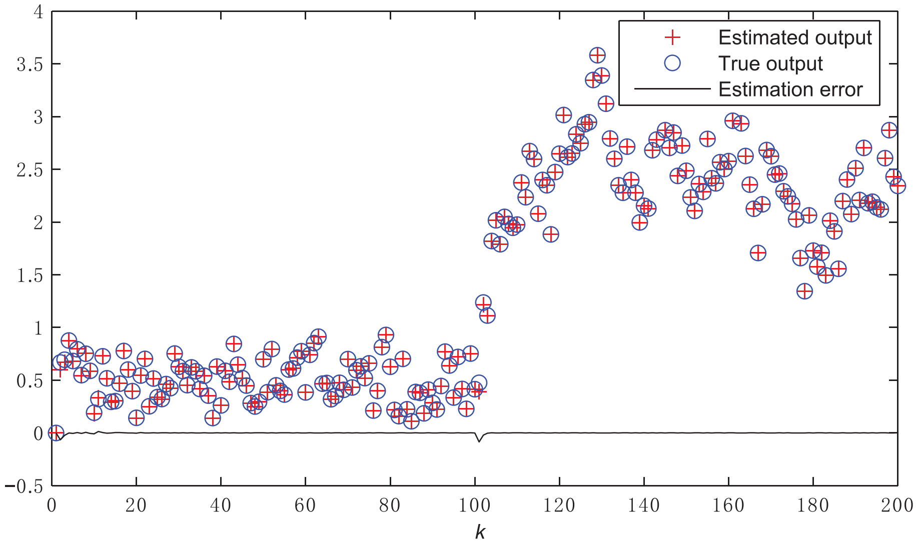

FIGURE 8 The output estimates, the true outputs, and their errors (switching model)

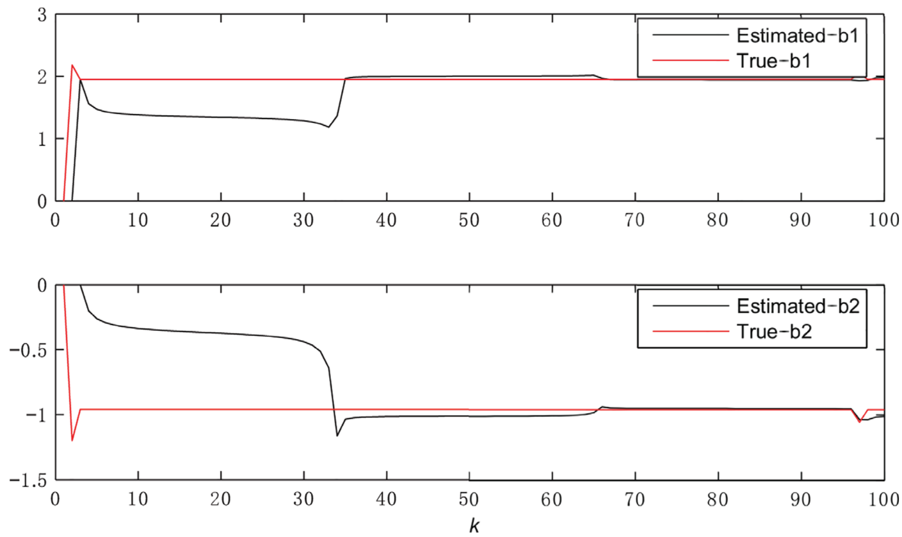

FIGURE 9 The true parameters and their estimates (ExpAR model)

Let

$$b_1(k) = 1.95 + 0.23e^{-o^2(k-1)}, \quad b_2(k) = -0.96 - 0.24e^{-o^2(k-1)}.$$

In the simulation, the noise $v(k)$ satisfies $v(k) \backsim N(0, 0.01)$. The parameter estimates, the output estimates, and the true values with their errors are shown in Figures 9 and 10, respectively ($\mu = 0.1$, $L = 100$).

Examples 1 and 2 show that the FLS estimates can catch the true values with a large weight μ when the parameters evolve slowly over time. When the parameters evolve intensively in a few sampling instants (Examples 3 and 4), for example, the switching system, the FLS estimates can asymptotically converge to the true values if k and $L - k$ are large enough, on the basis of a small weight μ.

7 | CONCLUSIONS

FLS algorithm is an efficient method for time-varying parameter systems whose parameters evolve slowly over time. In this article, an augmented FLS algorithm is developed with less computational efforts and simple structures. In addition,

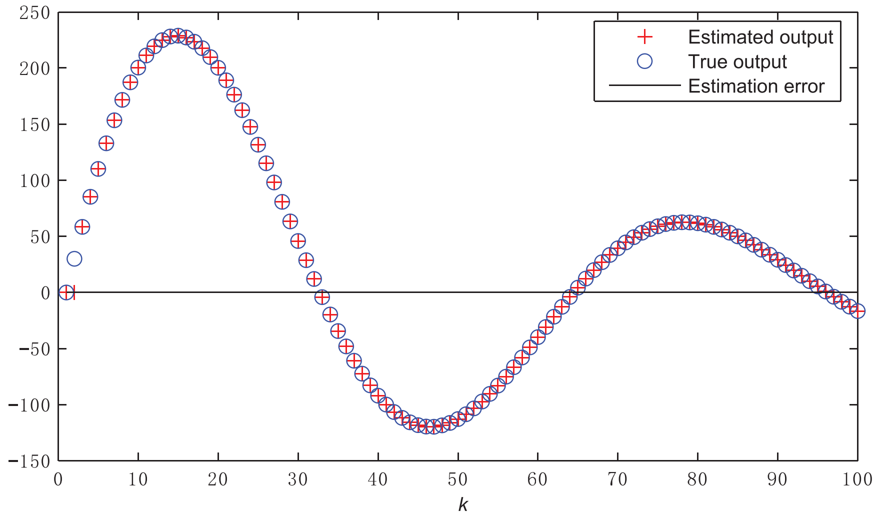

FIGURE 10 The output estimates, the true outputs, and their errors (ExpAR model)

this algorithm can be extended to polynomial nonlinear models, rational models, switching models, and exponential autoregressive models. Some properties of the augmented FLS algorithm are provided to help readers for their ad hoc research and applications. Simulation examples show that the proposed algorithm is effective.

This algorithm has introduced some interesting topics and given some new ideas in system identification. For example, it can be applied to non-stable system identification and can identify variable structure systems. The features will have positive impact to academic research and industrial applications in control engineering as well.

ACKNOWLEDGMENTS

The authors would like to express their gratitude to the editors and anonymous reviewers for their helpful comments and constructive suggestions regarding the revision of this article. This work is supported by the National Natural Science Foundation of China (No. 61973137), the Natural Science Foundation of Jiangsu Province (No. BK20201339), and the Funds of the Science and Technology on Near-Surface Detection Laboratory (Nos. 61424140207, 61424140202).

CONFLICT OF INTEREST

The authors declared that they have no conflicts of interest to this work.

DATA AVAILABILITY STATEMENT

All data generated or analyzed during this study are included in this article.

ORCID

Jing Chen https://orcid.org/0000-0001-5615-2255
Manfeng Hu https://orcid.org/0000-0001-6169-3013
Quanmin Zhu https://orcid.org/0000-0001-8173-1179

REFERENCES

1. Kalaba R, Tesfatsion L. Time-varying linear regression via flexible least squares. *Comput Math Appl*. 1989;17(8):1215-1245.
2. Chen GY, Gan M. A regularized variable projection algorithm for separable nonlinear least squares problems. *IEEE Trans Automat Contr*. 2019;64(2):526-537.
3. Jiao M, Wang DQ, Qiu JL. A GRU-RNN based momentum optimized algorithm for SOC estimation. *J Power Sources*. 2020;459:228051.
4. Zhou F, Gan M, Chen CLP. State-dependent ARX model-based RPC with variable feedback control laws for output tracking. *IEEE Trans Ind Electron*. 2021;68(5):4228-4237.
5. Ding JL, Zhang WH. Finite-time adaptive control for nonlinear systems with uncertain parameters based on the command filters. *Int J Adapt Control Signal Process*. 2021;35(9):1754-1767.
6. Ding F, Zhang X. The innovation algorithms for multivariable state-space models. *Int J Adapt Control Signal Process*. 2019;33(11):1601-1608.

7. Bottegal G, Aravkin AY, Hjalmarsson H, Pillonetto G. Robust EM kernel-based methods for linear system identification. *Automatica*. 2016;67:114-126.
8. Yu CP, Ljung L, Wills A, Verhaegen M. Constrained subspace method for the identification of structured state-space models. *IEEE Trans Automat Contr*. 2020;65(10):4201-4214.
9. Xu L, Ding F, Yang EF. Auxiliary model multi-innovation stochastic gradient parameter estimation methods for nonlinear sandwich systems. *Int J Robust Nonlinear Control*. 2021;31(1):148-165.
10. Zhang X. Adaptive parameter estimation for a general dynamical system with unknown states. *Int J Robust Nonlinear Control*. 2020;30(4):1351-1372.
11. Ji Y, Kang Z. Three-stage forgetting factor stochastic gradient parameter methods for a class of nonlinear systems. *Int J Robust Nonlinear Control*. 2021;31(3):971-987.
12. Chen MT, Ding F, Lin R, Ng TY, Zhang Y, Wei W. Maximum likelihood least squares-based iterative methods for output-error bilinear-parameter models with colored noises. *Int J Robust Nonlinear Control*. 2020;30(15):6262-6280.
13. Fan YM, Liu XM. Two-stage auxiliary model gradient-based iterative algorithm for the input nonlinear controlled autoregressive system with variable-gain nonlinearity. *Int J Robust Nonlinear Control*. 2020;30(14):5492-5509.
14. Wang DQ, Fan QH, Ma Y. An interactive maximum likelihood estimation method for multivariable Hammerstein systems. *J Frankl Inst*. 2020;357(17):12986-13005.
15. Fei QL, Ma JX, Xiong WL, Guo F. Variational Bayesian identification for bilinear state space models with Markov-switching time delays. *Int J Robust Nonlinear Control*. 2020;30(17):7478-7495.
16. Ding F, Lv L, Pan J, Wan X, Jin XB. Two-stage gradient-based iterative estimation methods for controlled autoregressive systems using the measurement data. *Int J Control Autom Syst*. 2020;18(4):886-896.
17. Ding F, Xu L, Zhu QM. Performance analysis of the generalised projection identification for time-varying systems. *IET Control Theory Appl*. 2016;10(18):2506-2514.
18. Yang XQ, Yang XB. Local identification of LPV dual-rate system with random measurement delays. *IEEE Trans Ind Electron*. 2018;65(2):1499-1507.
19. Yang X, Yin S. Robust global identification and output estimation for LPV dual-rate systems subjected to random output time-delays. *IEEE Trans Ind Inf*. 2017;13(6):2876-2885.
20. Lu YJ, Huang B, Khatibisepehr S. A variational Bayesian approach to robust identification of switched ARX models. *IEEE Trans Cybern*. 2015;46(12):2542-2547.
21. Kalaba R, Tesfatsion L. Flexible least squares for approximately linear systems. *IEEE Trans Syst Man Cybern*. 1990;20(5):978-987.
22. Montana G, Triantafyllopoulos K, Tsagaris T. Flexible least squares for temporal data mining and statistical arbitrage. *Expert Syst Appl*. 2009;36:2819-2830.
23. Ma JX, Huang B. Iterative identification of Hammerstein parameter varying systems with parameter uncertainties based on the variational Bayesian approach. *IEEE Trans Syst Man Cybern-Syst*. 2020;50(3):1035-1045.
24. Mu BQ, Bai EW, Zheng WX, Zhu Q. A globally consistent nonlinear least squares estimator for identification of nonlinear rational systems. *Automatica*. 2017;77:322-335.
25. Saxen JE, Saxen H, Toivonen HT. Identification of switching linear systems using self-organizing models with application to silicon prediction in hot metal. *Appl Soft Comput*. 2016;47:271-280.
26. Gan M, Chen GY, Chen L, Chen CL. Term selection for a class of nonlinear separable models. *IEEE Trans Neur Net Lear Syst*. 2020;31(2):445-451.
27. Wang DQ, Li LW, Ji Y, Yan Y. Model recovery for Hammerstein systems using the auxiliary model based orthogonal matching pursuit method. *Appl Math Model*. 2018;54:537-550.
28. Billings SA, Zhu QM. Rational model identification using extended least squares algorithm. *Int J Control*. 1991;54(3):529-546.
29. Chen J, Zhu QM, Li J, Liu YJ. Biased compensation recursive least squares-based threshold algorithm for time-delay rational models via redundant rule. *Nonlinear Dyn*. 2018;91(2):797-807.
30. Roy K, Kar S, Das RN. *Understanding the Basics of QSAR for Applications in Pharmaceutical Sciences and Risk Assessment*. Academic Press; 2015.
31. Chen J, Huang B, Ding F, Gu Y. Variational Bayesian approach for ARX systems with missing observations and varying time-delays. *Automatica*. 2018;94:194-204.
32. Xu H, Ding F, Yang EF. Three-stage multi-innovation parameter estimation for an exponential autoregressive time-series model with moving average noise by using the data filtering technique. *Int J Robust Nonlinear Control*. 2021;31(1):166-184.
33. Yang X, Huang B, Gao H. A direct maximum likelihood optimization approach to identification of LPV time-delay systems. *J Frankl Inst*. 2016;353(8):1862-1881.
34. Liu XP, Yang XQ. Robust variational inference for LPV dual-rate systems with randomly delayed outputs. *IEEE Trans Instrum Meas*. 2021;70:3001109. doi:10.1109/TIM.2021.3067242
35. Simon D. *Optimal State Estimation: Kalman, H_∞, and Nonlinear Approaches*. John Wiley & Sons, Inc; 2006.
36. Mao YW, Liu S, Liu JF. Robust economic model predictive control of nonlinear networked control systems with communication delays. *Int J Adapt Control Signal Process*. 2020;34(5):614-637.
37. Chen J, Shen QY, Ma JX, Liu YJ. Stochastic average gradient algorithm for multirate FIR models with varying time delays using self-organizing maps. *Int J Adapt Control Signal Process*. 2020;34(7):955-970.

38. Wang DQ, Zhang S, Gan M, Qiu J. A novel EM identification method for Hammerstein systems with missing output data. *IEEE Trans Ind Inf*. 2020;16(4):2500-2508.
39. Zhang X. Recursive parameter estimation and its convergence for bilinear systems. *IET Control Theory Appl*. 2020;14(5):677-688.
40. Xu L. Separable multi-innovation stochastic gradient estimation algorithm for the nonlinear dynamic responses of systems. *Int J Adapt Control Signal Process*. 2020;34(7):937-954.
41. Xu L, Chen FY. Hierarchical recursive signal modeling for multi-frequency signals based on discrete measured data. *Int J Adapt Control Signal Process*. 2021;35(5):676-693.
42. Zhu QM. A back propagation algorithm to estimate the parameters of nonlinear dynamic rational models. *Appl Math Model*. 2003;27(3):169-187.
43. Chen GY, Gan M. Generalized exponential autoregressive models for nonlinear time series: stationarity, estimation and applications. *Estimat Appl Inf Sci*. 2018;438:46-57.

How to cite this article: Chen J, Guo L, Hu M, Gan M, Zhu Q. Augmented flexible least squares algorithm for time-varying parameter systems. *Int J Robust Nonlinear Control*. 2021;1-19. doi: 10.1002/rnc.5972