Gradient-based iterative parameter estimation for bilinear-in-parameter systems using the model decomposition technique

Mengting Chen[1], Feng Ding[*,1,2], Erfu Yang[3]

1. Laboratory of Advanced Process Control for Light Industry (Ministry of Education), School of Internet of Things Engineering, Jiangnan University, Wuxi 214122, People's Republic of China
2. College of Automation and Electronic Engineering, Qingdao University of Science and Technology, Qingdao 266061, PR China
3. Space Mechatronic Systems Technology Laboratory, University of Strathclyde, Glasgow G1 1XJ, Scotland, United Kingdom
* E-mail: fding@jiangnan.edu.cn

Abstract: This paper studies the parameter estimation issues of a block-oriented nonlinear system that is bilinear in the parameters, i.e., the bilinear-in-parameter system. Using the model decomposition technique, the bilinear-in-parameter model is decomposed into two fictitious submodels: one containing the unknown parameters in the non-linear block and the other containing the unknown parameters in the linear dynamic one and the noise model. Then a gradient-based iterative algorithm is proposed to estimate all the unknown parameters by formulating and minimizing two criterion functions. The stochastic gradient algorithms is provided for comparison. The simulation results indicate that the proposed iterative algorithm can give higher parameter estimation accuracy than the stochastic gradient algorithms.

1 Introduction

Parameter estimation plays a major role in signal processing [1, 2], control system design [3–6]. The bilinear-in-parameter model can be used to describe the block-oriented nonlinear system, such as Hammerstein systems [7], Wiener systems [8, 9] and their combinations [10]. Much effort has been devoted to the parameter estimation for above block-oriented nonlinear systems [11, 12]. For example, a three-stage recursive least squares scheme was derived to estimate Hammerstein systems, which consist of the interaction of a static nonlinear function and a linear dynamic subsystem [13]; a two-experiment method was proposed to identify Wiener systems with a sine signal using the measured input-output data acquired from two separate experiments [14]; introducing the auxiliary model based identification idea, a modified multi-innovation gradient algorithm for Wiener-Hammerstein systems with unknown orders linear subsystems and backlash was developed based on the key-term separation principle [15].

Parameter estimation methods can be applied to many areas [16–19]. Parameter estimation algorithms can be utilized to obtain parameter estimates by fitting the measurable information to a befitting model structure in an optimal way [20]. Various parameter estimation algorithms such as recursive methods [21, 22], iterative methods [23, 24] and multi-innovation methods [25, 26] were presented to identify the systems in complex industrial processes. Recently, employing the hierarchical identification principle, Li and Liu proposed an auxiliary model based least squares iterative algorithm and a filtering based least squares iterative algorithm for bilinear systems [27, 28]; about the decomposition based identification, Xu and Ding discussed the parameter estimation algorithms

for multi-frequency signals based on the gradient search and the hierarchical identification principle [29].

The bilinear-in-parameter systems have been widely applied to various areas, such as digital filter synthesis, active noise control and neural networks since they can approximate the block-oriented nonlinear systems and behave similarly to linear models to some extent [30, 31]. Wang et al. transformed a multi-input multi-output Hammerstein system into a bilinear-in-parameter model, and proposed a hierarchical extended stochastic gradient algorithm [32]; a hierarchical least squares approach has been proposed for a class of bilinear-in-parameter systems [33]. Another algorithm of identifying bilinear-in-parameter systems is the over-parameterization method [34]. However, the resulting over-parameterization model exists the cross-product terms of the parameters, leading to many redundant parameter estimates and heavy computational burden [35].

The iterative methods are usually used to estimate the parameters of linear and nonlinear systems in which the information vector contains unknown variables. The decomposition technique can convert a system into several subsystems with small dimensions [36–38]. On the basis of the work in [39], this paper discusses the parameter estimation methods for bilinear-in-parameter systems by using the recursive idea and the iterative technique. Different from the work in [39], in order to avoid the redundant estimates caused by the over-parameterization method, we use the model decomposition technique to construct the two identification models. The main contributions of this paper are as follows.

- By using the negative gradient search and the multi-innovation identification theory, a decomposition based generalized extended

stochastic gradient algorithm and a decomposition based multi-innovation generalized extended stochastic gradient algorithm are presented for the bilinear-in-parameter system.
- By using a fixed data batch with a finite length, a decomposition based generalized extended gradient iterative algorithm is derived for generating more accurate parameter estimates than the stochastic gradient algorithms.
- By using the model decomposition technique, the original system is decomposed into two subsystems, and the proposed parameter estimation algorithms do not involve the over-parameterization of the parameters, and avoid estimating redundant parameters.

The outlines of this paper are organized as follows. Section 2 derives the identification models of the bilinear-in-parameter system with colored noise. For comparison, Section 3 presents two stochastic gradient algorithms. The decomposition based generalized extended gradient iterative algorithm is derived in Section 4. Section 5 offers an example to illustrate the effectiveness of the proposed algorithm. Finally, some concluding remarks are offered in Section 6.

2 The identification model for bilinear-in-parameter systems

First of all, let us introduce some notation. "$A =: X$" or "$X := A$" stands for "A is defined as X"; the symbol $\boldsymbol{I}$ ($\boldsymbol{I}_n$) represents an identity matrix of appropriate size ($n \times n$); $\boldsymbol{1}_n$ stands for an n-dimensional column vector whose elements are all 1, $\boldsymbol{1}_{m \times n}$ represents a matrix of size $m \times n$ whose elements are all 1; the superscript T denotes the matrix/vector transpose; $\lambda_{\max}[\boldsymbol{X}]$ denotes the maximum eigenvalue of symmetric square matrix $\boldsymbol{X}$; the norm of a matrix $\boldsymbol{X}$ is defined by $\|\boldsymbol{X}\|^2 := \mathrm{tr}[\boldsymbol{X}\boldsymbol{X}^{\mathrm{T}}]$.

Consider the following bilinear-in-parameter system,

$$y(t) = \boldsymbol{a}^{\mathrm{T}}\boldsymbol{F}(t)\boldsymbol{b} + w(t), \quad (1)$$

$$\boldsymbol{F}(t) = [\boldsymbol{f}(u(t-1)), \boldsymbol{f}(u(t-2)), \cdots, \boldsymbol{f}(u(t-n))], \quad (2)$$

$$\boldsymbol{f}(u(t)) = [f_1(u(t)), f_2(u(t)), \cdots, f_m(u(t))]^{\mathrm{T}}, \quad (3)$$

where $y(t) \in \mathbb{R}$ is the system output, $\boldsymbol{F}(t) \in \mathbb{R}^{m \times n}$ is composed of available measurement data, $u(t) \in \mathbb{R}$ is the system input, $\boldsymbol{f}(u(t)) \in \mathbb{R}^m$ is the known nonlinear basis function, which may be polynomials or trigonometric functions, $\boldsymbol{a} := [a_1, a_2, \cdots, a_m]^{\mathrm{T}} \in \mathbb{R}^m$ and $\boldsymbol{b} := [b_1, b_2, \cdots, b_n]^{\mathrm{T}} \in \mathbb{R}^n$ are the unknown parameter vectors to be estimated.

Remark 1. For the bilinear-in-parameter system with colored noise, the disturbance with zero mean $w(t) \in \mathbb{R}$ includes three special cases: 1) $w(t) := \frac{1}{C(z)}v(t)$ is an autoregressive (AR) process [40]; 2) $w(t) := D(z)v(t)$ is a moving average (MA) process [41]; 3) $w(t) := \frac{D(z)}{C(z)}v(t)$ is an autoregressive moving average (ARMA) process [42]; $v(t) \in \mathbb{R}$ is a random white noise with zero mean and variance σ^2, $C(z)$ and $D(z)$ are polynomials in the unit backward shift operator z^{-1} ($z^{-1}y(t) = y(t-1)$).

Without loss of generality, assume that the orders m, n, n_c and n_d are predetermined, and $y(t) = 0$, $u(t) = 0$, $w(t) = 0$ and $v(t) = 0$ for $t \leq 0$. This paper considers the iterative algorithms for the bilinear-in-parameter system with ARMA noise. Then, Equation (1) can be written as

$$y(t) = \boldsymbol{a}^{\mathrm{T}}\boldsymbol{F}(t)\boldsymbol{b} + \frac{D(z)}{C(z)}v(t), \quad (4)$$

$$C(z) = 1 + c_1 z^{-1} + c_2 z^{-2} + \cdots + c_{n_c} z^{-n_c},$$

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

In order to avoid estimating the redundant parameters, we use the model decomposition to express the system output as a linear combination of the system parameters.

Define the system parameter vectors $\boldsymbol{\rho}$ and $\boldsymbol{\vartheta}$, the information vectors $\boldsymbol{\phi}_1(t)$, $\boldsymbol{\phi}_2(t)$, $\boldsymbol{\varphi}(t)$ and $\boldsymbol{\psi}(t)$ and the intermediate variable $\xi(t)$ as

$$\boldsymbol{\rho} := [c_1, c_2, \cdots, c_{n_c}, d_1, d_2, \cdots, d_{n_d}]^{\mathrm{T}} \in \mathbb{R}^{n_c + n_d}, \quad (5)$$

$$\boldsymbol{\vartheta} := \begin{bmatrix} \boldsymbol{b} \\ \boldsymbol{\rho} \end{bmatrix} \in \mathbb{R}^{n + n_c + n_d},$$

$$\boldsymbol{\phi}_1(t) := \boldsymbol{F}(t)\boldsymbol{b} \in \mathbb{R}^m, \quad \boldsymbol{\phi}_2(t) := \boldsymbol{F}^{\mathrm{T}}(t)\boldsymbol{a} \in \mathbb{R}^n,$$

$$\boldsymbol{\varphi}(t) := [-w(t-1), -w(t-2), \cdots, -w(t-n_c),$$
$$v(t-1), v(t-2), \cdots, v(t-n_d)]^{\mathrm{T}} \in \mathbb{R}^{n_c + n_d}, \quad (6)$$

$$\boldsymbol{\psi}(t) := \begin{bmatrix} \boldsymbol{\phi}_2(t) \\ \boldsymbol{\varphi}(t) \end{bmatrix} \in \mathbb{R}^{n + n_c + n_d},$$

$$\xi(t) := y(t) - \boldsymbol{\varphi}^{\mathrm{T}}(t)\boldsymbol{\rho} \in \mathbb{R}. \quad (7)$$

From (5) and (6), we have

$$w(t) = \frac{D(z)}{C(z)}v(t)$$
$$= [1 - C(z)]w(t) + D(z)v(t)$$
$$= -\sum_{i=1}^{n_c} c_i w(t-i) + \sum_{i=1}^{n_d} d_i v(t-i) + v(t)$$
$$= \boldsymbol{\varphi}^{\mathrm{T}}(t)\boldsymbol{\rho} + v(t). \quad (8)$$

From the above definitions, Equations (4) can be equivalently written as

$$y(t) = \boldsymbol{a}^{\mathrm{T}}\boldsymbol{F}(t)\boldsymbol{b} + w(t)$$
$$= \boldsymbol{\phi}_1^{\mathrm{T}}(t)\boldsymbol{a} + w(t) \quad (9)$$
$$= \boldsymbol{\phi}_1^{\mathrm{T}}(t)\boldsymbol{a} + \boldsymbol{\varphi}^{\mathrm{T}}(t)\boldsymbol{\rho} + v(t) \quad (10)$$
$$= \boldsymbol{\phi}_2^{\mathrm{T}}(t)\boldsymbol{b} + \boldsymbol{\varphi}^{\mathrm{T}}(t)\boldsymbol{\rho} + v(t)$$
$$= \boldsymbol{\psi}^{\mathrm{T}}(t)\boldsymbol{\vartheta} + v(t). \quad (11)$$

From (7) and (10), we have

$$\xi(t) = \boldsymbol{\phi}_1^{\mathrm{T}}(t)\boldsymbol{a} + v(t). \quad (12)$$

Equations (11)–(12) form the identification models for the bilinear-in-parameter system in (4) based on the decomposition. Considering the identifiability of the bilinear-in-parameter system in (4), two identification schemes are discussed: one is the recursive parameter estimation method and the other is iterative parameter estimation method. In this paper, we use the recursive index t, e.g., $\hat{\boldsymbol{\theta}}(t)$ as the parameter estimate for the stochastic gradient algorithm, and the iterative index with subscript k, e.g., $\hat{\boldsymbol{\theta}}_k$ to denote the parameter estimate for the gradient-based iterative algorithm.

3 The stochastic gradient algorithms

For the purpose of showing the effectiveness of the proposed algorithm, we derive a stochastic gradient algorithm as the basic algorithm in this section. According to the parameter identification models in (11)–(12), we define two gradient criterion functions:

$$J_1(\boldsymbol{a}) := \frac{1}{2}[\xi(t) - \boldsymbol{\phi}_1^{\mathrm{T}}(t)\boldsymbol{a}]^2,$$

$$J_2(\boldsymbol{\vartheta}) := \frac{1}{2}[y(t) - \boldsymbol{\psi}^{\mathrm{T}}(t)\boldsymbol{\vartheta}]^2.$$

Using the negative search and minimizing the criterion functions $J_1(\boldsymbol{a})$ and $J_2(\boldsymbol{\vartheta})$ give

$$\hat{\boldsymbol{a}}(t) = \hat{\boldsymbol{a}}(t-1) - \frac{1}{r_1(t)}\mathrm{grad}[J_1(\boldsymbol{a}(t-1))]$$

$$= \hat{\boldsymbol{a}}(t-1) + \frac{\boldsymbol{\phi}_1(t)}{r_1(t)}[\xi(t) - \boldsymbol{\phi}_1^{\mathrm{T}}(t)\hat{\boldsymbol{a}}(t-1)]$$

$$= \hat{\boldsymbol{a}}(t-1) + \frac{\boldsymbol{\phi}_1(t)}{r_1(t)}$$
$$\times [y(t) - \boldsymbol{\varphi}^{\mathrm{T}}(t)\boldsymbol{\rho} - \boldsymbol{\phi}_1^{\mathrm{T}}(t)\hat{\boldsymbol{a}}(t-1)], \quad (13)$$

$$r_1(t) = r_1(t-1) + \|\boldsymbol{\phi}_1(t)\|^2, \quad r_1(0) = 1, \quad (14)$$

$$\hat{\boldsymbol{\vartheta}}(t) = \hat{\boldsymbol{\vartheta}}(t-1) - \frac{1}{r_2(t)}\mathrm{grad}[J_2(\boldsymbol{\vartheta}(t-1))]$$

$$= \hat{\boldsymbol{\vartheta}}(t-1) + \frac{\boldsymbol{\psi}(t)}{r_2(t)}[y(t) - \boldsymbol{\psi}^{\mathrm{T}}(t)\hat{\boldsymbol{\vartheta}}(t-1)], \quad (15)$$

$$r_2(t) = r_2(t-1) + \|\boldsymbol{\psi}(t)\|^2, \quad r_2(0) = 1, \quad (16)$$

where $1/r_1(t)$ and $1/r_2(t)$ are the step-sizes i.e., the convergence factors. Since the right-hand sides of (13)–(16) contain the unknown parameter vectors $\boldsymbol{a}$, $\boldsymbol{b}$ and $\boldsymbol{\rho}$ and the unmeasurable noise terms $w(t-i)$ and $v(t-i)$, the above algorithm is impossible to implement. The solution is to replace $\boldsymbol{a}$, $\boldsymbol{b}$ and $\boldsymbol{\rho}$ with their preceding estimates $\hat{\boldsymbol{a}}(t-1)$, $\hat{\boldsymbol{b}}(t-1)$ and $\hat{\boldsymbol{\rho}}(t-1)$, and to replace $w(t-i)$ and $v(t-i)$ with their current estimates $\hat{w}(t-i)$ and $\hat{v}(t-i)$, and use them to define the information vectors $\hat{\boldsymbol{\phi}}_1(t)$, $\hat{\boldsymbol{\phi}}_2(t)$ and $\hat{\boldsymbol{\psi}}(t)$ and the parameter estimation vector $\hat{\boldsymbol{\vartheta}}(t)$ as

$$\hat{\boldsymbol{\phi}}_1(t) := \boldsymbol{F}(t)\hat{\boldsymbol{b}}(t-1) \in \mathbb{R}^m, \quad \hat{\boldsymbol{\phi}}_2(t) := \boldsymbol{F}^{\mathrm{T}}(t)\hat{\boldsymbol{a}}(t-1) \in \mathbb{R}^n,$$

$$\hat{\boldsymbol{\psi}}(t) := \begin{bmatrix} \hat{\boldsymbol{\phi}}_2(t) \\ \hat{\boldsymbol{\varphi}}(t) \end{bmatrix} \in \mathbb{R}^{n+n_c+n_d}, \quad \hat{\boldsymbol{\vartheta}}(t) := \begin{bmatrix} \hat{\boldsymbol{b}}(t) \\ \hat{\boldsymbol{\rho}}(t) \end{bmatrix} \in \mathbb{R}^{n+n_c+n_d}.$$

Replacing the parameter vectors $\boldsymbol{a}$ and $\boldsymbol{\rho}$ and the information vectors $\boldsymbol{\phi}_1(t)$ and $\boldsymbol{\varphi}(t)$ in (8) and (9) with their estimates $\hat{\boldsymbol{a}}(t)$, $\hat{\boldsymbol{\rho}}(t)$, $\hat{\boldsymbol{\phi}}_1(t)$ and $\hat{\boldsymbol{\varphi}}(t)$, the estimates $\hat{w}(t)$ and $\hat{v}(t)$ can be computed through

$$\hat{w}(t) := y(t) - \hat{\boldsymbol{\phi}}_1^{\mathrm{T}}(t)\hat{\boldsymbol{a}}(t),$$
$$\hat{v}(t) := \hat{w}(t) - \hat{\boldsymbol{\varphi}}^{\mathrm{T}}(t)\hat{\boldsymbol{\rho}}(t).$$

According to the above derivation, we yield the following the decomposition based generalized extended stochastic gradient (D-GESG) algorithm:

$$\bar{\boldsymbol{a}}(t) = \hat{\boldsymbol{a}}(t-1) + \frac{\hat{\boldsymbol{\phi}}_1(t)}{r_1(t)}$$
$$\times [y(t) - \hat{\boldsymbol{\phi}}_1^{\mathrm{T}}(t)\hat{\boldsymbol{a}}(t-1) - \hat{\boldsymbol{\varphi}}^{\mathrm{T}}(t)\hat{\boldsymbol{\rho}}(t-1)], \quad (17)$$

$$r_1(t) = r_1(t-1) + \|\hat{\boldsymbol{\phi}}_1(t)\|^2, \quad r_1(0) = 1, \quad (18)$$

$$\hat{\boldsymbol{\vartheta}}(t) = \hat{\boldsymbol{\vartheta}}(t-1) + \frac{\hat{\boldsymbol{\psi}}(t)}{r_2(t)}[y(t) - \hat{\boldsymbol{\psi}}^{\mathrm{T}}(t)\hat{\boldsymbol{\vartheta}}(t-1)], \quad (19)$$

$$r_2(t) = r_2(t-1) + \|\hat{\boldsymbol{\psi}}(t)\|^2, \quad r_2(0) = 1, \quad (20)$$

$$\hat{\boldsymbol{\phi}}_1(t) = \boldsymbol{F}(t)\hat{\boldsymbol{b}}(t-1), \quad \hat{\boldsymbol{\phi}}_2(t) = \boldsymbol{F}^{\mathrm{T}}(t)\hat{\boldsymbol{a}}(t-1), \quad (21)$$

$$\hat{\boldsymbol{\varphi}}(t) = [-\hat{w}(t-1), -\hat{w}(t-2), \cdots, -\hat{w}(t-n_c),$$
$$\hat{v}(t-1), \hat{v}(t-2), \cdots, \hat{v}(t-n_d)]^{\mathrm{T}}, \quad (22)$$

$$\hat{w}(t) = y(t) - \hat{\boldsymbol{\phi}}_1^{\mathrm{T}}(t)\hat{\boldsymbol{a}}(t), \quad (23)$$

$$\hat{v}(t) = \hat{w}(t) - \hat{\boldsymbol{\varphi}}^{\mathrm{T}}(t)\hat{\boldsymbol{\rho}}(t). \quad (24)$$

Remark 2. The bilinear-in-parameter system $y(t) = \boldsymbol{a}^{\mathrm{T}}\boldsymbol{F}(t)\boldsymbol{b} + w(t)$ contains the product term of the parameter vectors $\boldsymbol{a}$ and $\boldsymbol{b}$. To have identifiability, we adopt the normalization constraint on $\boldsymbol{a}$ for the system in (4) [43]. Assume that $\|\boldsymbol{a}\| = 1$, and the first nonzero element of the parameter vector $\boldsymbol{a}$ is positive, i.e., $a_1 > 0$. Then we have

$$\hat{\boldsymbol{a}}(t) = \mathrm{sgn}[\bar{a}_1(t)] \frac{\bar{\boldsymbol{a}}(t)}{\|\bar{\boldsymbol{a}}(t)\|}, \quad (25)$$

where $\mathrm{sgn}[\bar{a}_1(t)]$ denotes the sign of $\bar{a}_1(t)$.

Remark 3. The D-GESG algorithm can be on-line implemented. The initial values are taken to be $\hat{\boldsymbol{a}}(0) = \mathbf{1}_m/\sqrt{m}$ (to guarantee that $\|\hat{\boldsymbol{a}}(0)\| = 1$), $\hat{\boldsymbol{\vartheta}}(0) = \mathbf{1}_{n+n_c+n_d}/p_0$, $\hat{w}(t-i) = 1/p_0$ and $\hat{v}(t-i) = 1/p_0$, $i = 1, 2, \cdots, \max[n_c, n_d]$, where p_0 is a large number, e.g., $p_0 = 10^6$.

In order to improve the parameter estimation accuracy of the D-GESG algorithm, we study another stochastic gradient algorithm for bilinear-in-parameter systems by using the multi-innovation theory [44]. Let $n_1 := m + n_c + n_d$, $n_2 := n + n_c + n_d$, $n_0 := m + n + n_c + n_d$ and p represents the innovation length. Consider p data from $j = t - p + 1$ to $j = t$ to define the stacked output vector $\boldsymbol{Y}(p, t)$ and the stacked information matrices $\boldsymbol{\Phi}_1(p, t)$, $\boldsymbol{\Psi}(p, t)$ and $\boldsymbol{\Phi}_n(p, t)$ as

$$\boldsymbol{Y}(p, t) := [y(t), y(t-1), \cdots, y(t-p+1)]^{\mathrm{T}} \in \mathbb{R}^p,$$

$$\boldsymbol{\Phi}_1(p, t) := [\boldsymbol{\phi}_1(t), \boldsymbol{\phi}_1(t-1), \cdots, \boldsymbol{\phi}_1(t-p+1)]$$
$$= [\boldsymbol{F}(t)\boldsymbol{b}, \boldsymbol{F}(t-1)\boldsymbol{b}, \cdots, \boldsymbol{F}(t-p+1)\boldsymbol{b}] \in \mathbb{R}^{m \times p},$$

$$\boldsymbol{\Psi}(p, t) := [\boldsymbol{\psi}(t), \cdots, \boldsymbol{\psi}(t-p+1)]$$
$$= \begin{bmatrix} \boldsymbol{F}^{\mathrm{T}}(t)\boldsymbol{a} & \cdots & \boldsymbol{F}^{\mathrm{T}}(t-p+1)\boldsymbol{a} \\ \boldsymbol{\varphi}(t) & \cdots & \boldsymbol{\varphi}(t-p+1) \end{bmatrix} \in \mathbb{R}^{n_2 \times p},$$

$$\boldsymbol{\Phi}_n(p, t) := [\boldsymbol{\varphi}(t), \boldsymbol{\varphi}(t-1), \cdots, \boldsymbol{\varphi}(t-p+1)] \in \mathbb{R}^{(n_c+n_d) \times p}.$$

From (17) and (19), define two scalar innovations

$$e_1(t) := y(t) - \hat{\boldsymbol{\phi}}_1^{\mathrm{T}}(t)\hat{\boldsymbol{a}}(t-1) - \hat{\boldsymbol{\varphi}}^{\mathrm{T}}(t)\hat{\boldsymbol{\rho}}(t-1)$$
$$= \hat{\xi}(t) - \hat{\boldsymbol{\phi}}_1^{\mathrm{T}}(t)\hat{\boldsymbol{a}}(t-1),$$

$$e_2(t) := y(t) - \hat{\boldsymbol{\psi}}^{\mathrm{T}}(t)\hat{\boldsymbol{\vartheta}}(t-1).$$

Then, we expend $e_1(t)$ and $e_2(t)$ into two innovation vectors $\boldsymbol{E}_1(p, t)$ and $\boldsymbol{E}_2(p, t)$ as

$$\boldsymbol{E}_1(p, t) := [e_1(t), e_1(t-1), \cdots, e_1(t-p+1)]^{\mathrm{T}} \in \mathbb{R}^p$$
$$= \begin{bmatrix} \hat{\xi}(t) - \hat{\boldsymbol{\phi}}_1^{\mathrm{T}}(t)\hat{\boldsymbol{a}}(t-1) \\ \hat{\xi}(t-1) - \hat{\boldsymbol{\phi}}_1^{\mathrm{T}}(t-1)\hat{\boldsymbol{a}}(t-2) \\ \vdots \\ \hat{\xi}(t-p+1) - \hat{\boldsymbol{\phi}}_1^{\mathrm{T}}(t-p+1)\hat{\boldsymbol{a}}(t-p) \end{bmatrix}, \quad (26)$$

$$\boldsymbol{E}_2(p, t) := [e_2(t), e_2(t-1), \cdots, e_2(t-p+1)]^{\mathrm{T}} \in \mathbb{R}^p$$
$$= \begin{bmatrix} y(t) - \hat{\boldsymbol{\psi}}^{\mathrm{T}}(t)\hat{\boldsymbol{\vartheta}}(t-1) \\ y(t-1) - \hat{\boldsymbol{\psi}}^{\mathrm{T}}(t-1)\hat{\boldsymbol{\vartheta}}(t-2) \\ \vdots \\ y(t-p+1) - \hat{\boldsymbol{\psi}}^{\mathrm{T}}(t-p+1)\hat{\boldsymbol{\vartheta}}(t-p) \end{bmatrix}. \quad (27)$$

Normally, one thinks that the estimates $\hat{\boldsymbol{a}}(t-1)$, $\hat{\boldsymbol{\rho}}(t-1)$ and $\hat{\boldsymbol{\vartheta}}(t-1)$ at time $(t-1)$ are closer to the true values $\boldsymbol{a}$, $\boldsymbol{\rho}$ and $\boldsymbol{\vartheta}$ than the estimates $\hat{\boldsymbol{a}}(t-i)$, $\hat{\boldsymbol{\rho}}(t-i)$ and $\hat{\boldsymbol{\vartheta}}(t-i)$ at time $(t-i)$ $(i \geq 2)$. Here, replacing the terms $\hat{\boldsymbol{a}}(t-i)$, $\hat{\boldsymbol{\rho}}(t-i)$ and $\hat{\boldsymbol{\vartheta}}(t-i)$ $(i \geq 2)$ with $\hat{\boldsymbol{a}}(t-1)$, $\hat{\boldsymbol{\rho}}(t-1)$ and $\hat{\boldsymbol{\vartheta}}(t-1)$ in (26)–(27), $\boldsymbol{E}_1(p, t)$ and $\boldsymbol{E}_2(p, t)$ can be reasonably taken as

$$\boldsymbol{E}_1(p, t) := \begin{bmatrix} \hat{\xi}(t) - \hat{\boldsymbol{\phi}}_1^{\mathrm{T}}(t)\hat{\boldsymbol{a}}(t-1) \\ \hat{\xi}(t-1) - \hat{\boldsymbol{\phi}}_1^{\mathrm{T}}(t-1)\hat{\boldsymbol{a}}(t-1) \\ \vdots \\ \hat{\xi}(t-p+1) - \hat{\boldsymbol{\phi}}_1^{\mathrm{T}}(t-p+1)\hat{\boldsymbol{a}}(t-1) \end{bmatrix} \in \mathbb{R}^p$$
$$= \boldsymbol{Y}(p, t) - \hat{\boldsymbol{\Phi}}_1^{\mathrm{T}}(p, t)\hat{\boldsymbol{a}}(t-1) - \hat{\boldsymbol{\Phi}}_n^{\mathrm{T}}(p, t)\hat{\boldsymbol{\rho}}(t-1),$$

$$\boldsymbol{E}_2(p, t) := \begin{bmatrix} y(t) - \hat{\boldsymbol{\psi}}^{\mathrm{T}}(t)\hat{\boldsymbol{\vartheta}}(t-1) \\ y(t-1) - \hat{\boldsymbol{\psi}}^{\mathrm{T}}(t-1)\hat{\boldsymbol{\vartheta}}(t-1) \\ \vdots \\ y(t-p+1) - \hat{\boldsymbol{\psi}}^{\mathrm{T}}(t-p+1)\hat{\boldsymbol{\vartheta}}(t-1) \end{bmatrix} \in \mathbb{R}^p$$

$$= \boldsymbol{Y}(p,t) - \hat{\boldsymbol{\Psi}}^{\mathrm{T}}(p,t)\hat{\boldsymbol{\vartheta}}(t-1).$$

From the above definitions, we can obtain the following decomposition based multi-innovation generalized extended stochastic gradient (D-MI-GESG) algorithm:

$$\bar{\boldsymbol{a}}(t) = \hat{\boldsymbol{a}}(t-1) + \frac{\hat{\boldsymbol{\Phi}}_1(p,t)}{r_1(t)}\boldsymbol{E}_1(p,t), \tag{28}$$

$$\boldsymbol{E}_1(p,t) = \boldsymbol{Y}(p,t) - \hat{\boldsymbol{\Phi}}_1^{\mathrm{T}}(p,t)\hat{\boldsymbol{a}}(t-1) - \hat{\boldsymbol{\Phi}}_{\mathrm{n}}^{\mathrm{T}}(p,t)\hat{\boldsymbol{\rho}}(t-1) \tag{29}$$

$$r_1(t) = r_1(t-1) + \|\hat{\boldsymbol{\Phi}}_1(p,t)\|^2, \tag{30}$$

$$\hat{\boldsymbol{\vartheta}}(t) = \hat{\boldsymbol{\vartheta}}(t-1) + \frac{\hat{\boldsymbol{\Psi}}(p,t)}{r_2(t)}\boldsymbol{E}_2(p,t), \tag{31}$$

$$\boldsymbol{E}_2(p,t) = \boldsymbol{Y}(p,t) - \hat{\boldsymbol{\Psi}}^{\mathrm{T}}(p,t)\hat{\boldsymbol{\vartheta}}(t-1), \tag{32}$$

$$r_2(t) = r_2(t-1) + \|\hat{\boldsymbol{\Psi}}(p,t)\|^2, \tag{33}$$

$$\boldsymbol{Y}(p,t) = [y(t), y(t-1), \cdots, y(t-p+1)]^{\mathrm{T}}, \tag{34}$$

$$\hat{\boldsymbol{\Phi}}_1(p,t) = [\boldsymbol{F}(t)\hat{\boldsymbol{b}}(t-1), \cdots, \boldsymbol{F}(t-p+1)\hat{\boldsymbol{b}}(t-1)], \tag{35}$$

$$\hat{\boldsymbol{\Psi}}(p,t) := [\hat{\boldsymbol{\psi}}(t), \cdots, \hat{\boldsymbol{\psi}}(t-p+1)], \tag{36}$$

$$\hat{\boldsymbol{\Phi}}_{\mathrm{n}}(p,t) = [\hat{\boldsymbol{\varphi}}(t), \hat{\boldsymbol{\varphi}}(t-1), \cdots, \hat{\boldsymbol{\varphi}}(t-p+1)], \tag{37}$$

$$\hat{\boldsymbol{\varphi}}(t) = [-\hat{w}(t-1), -\hat{w}(t-2), \cdots, -\hat{w}(t-n_c),$$
$$\hat{v}(t-1), \hat{v}(t-2), \cdots, \hat{v}(t-n_d)]^{\mathrm{T}}, \tag{38}$$

$$\hat{\boldsymbol{\phi}}_1(t) = \boldsymbol{F}(t)\hat{\boldsymbol{b}}(t-1), \tag{39}$$

$$\hat{\boldsymbol{\psi}}(t) := \begin{bmatrix} \boldsymbol{F}^{\mathrm{T}}(t)\hat{\boldsymbol{a}}(t-1) \\ \hat{\boldsymbol{\varphi}}(t) \end{bmatrix}, \tag{40}$$

$$\hat{w}(t) = y(t) - \hat{\boldsymbol{\phi}}_1^{\mathrm{T}}(t)\hat{\boldsymbol{a}}(t), \tag{41}$$

$$\hat{v}(t) = \hat{w}(t) - \hat{\boldsymbol{\varphi}}^{\mathrm{T}}(t)\hat{\boldsymbol{\rho}}(t), \tag{42}$$

$$\hat{\boldsymbol{\vartheta}}(t) = [\hat{\boldsymbol{b}}^{\mathrm{T}}(t)\hat{\boldsymbol{\rho}}^{\mathrm{T}}(t)]^{\mathrm{T}}, \tag{43}$$

$$\hat{\boldsymbol{a}}(t) = \mathrm{sgn}[\bar{a}_1(t)]\frac{\bar{\boldsymbol{a}}(t)}{\|\bar{\boldsymbol{a}}(t)\|}. \tag{44}$$

The procedures for implementing the D-MI-GESG algorithm in (28)–(44) are as follows.

1. Let $t = 1$, choose an innovation length p, and set the initial values $\hat{\boldsymbol{a}}(0) = \mathbf{1}_m/\sqrt{m}$, $\hat{\boldsymbol{\vartheta}}(0) = \mathbf{1}_{n+n_c+n_d}/p_0$, $w(t-i) = 1/p_0$, $v(t-i) = 1/p_0$, $i = 1, 2, \cdots, \max[n_c + n_d]$, $p_0 = 10^6$ and give the nonlinear basis function $f_i(*)$.
2. Collect the measured data $y(t)$, $\boldsymbol{F}(t)$ and $\boldsymbol{Y}(p,t)$, and construct the information estimation vectors $\hat{\boldsymbol{\phi}}_1(t)$ and $\hat{\boldsymbol{\psi}}(t)$ using (39) and (40).
3. Compute $r_1(t)$ and $r_2(t)$ using (30) and (33).
4. Construct $\hat{\boldsymbol{\Phi}}_1(p,t)$, $\hat{\boldsymbol{\Psi}}(p,t)$ and $\hat{\boldsymbol{\Phi}}_{\mathrm{n}}(p,t)$ using (35), (36) and (37).
5. Compute the information vectors $\boldsymbol{E}_1(p,t)$ and $\boldsymbol{E}_2(p,t)$ using (29) and (32).
6. Update the parameter estimates $\hat{\boldsymbol{a}}(t)$ and $\hat{\boldsymbol{\vartheta}}(t)$ using (28) and (31) and normalize $\hat{\boldsymbol{a}}(t)$ using (44).
7. Compute $\hat{w}(t)$ using (41) and $\hat{v}(t)$ using (42).
8. Increase t by 1, and go to Step 2.

Remark 4. When the innovation length $p = 1$, we can obtain the D-GESG algorithm. Obviously, the D-MI-GESG algorithm has higher computational load than the D-GESG algorithm.

Remark 5. By introducing the innovation length p, the D-MI-GESG algorithm uses not only the current data information $\{u(t), y(t)\}$ and the innovations $e_j(t)$, $j = 1, 2$, but also the past data information $\{u(t-i), y(t-i), i = 1, 2, \cdots, p-1\}$ at each recursion and hence has higher parameter estimation accuracy than the D-GESG algorithm.

4   The gradient-based iterative algorithms

The D-GESG algorithm and the D-MI-GESG algorithm can estimate all the unknown parameters, but they use only the measured data $\{u(i), y(i), i = 1, 2, \cdots, t\}$. In other words, their weakness is that the algorithms do not make full use of all the system data information. This section develops a gradient-based iterative algorithm for the bilinear-in-parameter system in (4) based on the model decomposition technique.

Use the newest l data and define the stacked output vector $\boldsymbol{Y}(t)$ and the stacked information matrices $\boldsymbol{\Phi}_1(t)$, $\boldsymbol{\Psi}(t)$ and $\boldsymbol{\Phi}_{\mathrm{n}}(t)$ as

$$\boldsymbol{Y}(t) := [y(t), y(t-1), \cdots, y(t-l+1)]^{\mathrm{T}} \in \mathbb{R}^l, \tag{45}$$

$$\boldsymbol{\Phi}_1(t) := [\boldsymbol{\phi}_1(t), \cdots, \boldsymbol{\phi}_1(t-l+1)]$$
$$= [\boldsymbol{F}(t)\boldsymbol{b}, \cdots, \boldsymbol{F}(t-l+1)\boldsymbol{b}] \in \mathbb{R}^{m \times l}, \tag{46}$$

$$\boldsymbol{\Psi}(t) := [\boldsymbol{\psi}(t), \cdots, \boldsymbol{\psi}(t-l+1)]$$
$$= \begin{bmatrix} \boldsymbol{F}^{\mathrm{T}}(t)\boldsymbol{a} & \cdots & \boldsymbol{F}^{\mathrm{T}}(t-l+1)\boldsymbol{a} \\ \boldsymbol{\varphi}(t) & \cdots & \boldsymbol{\varphi}(t-l+1) \end{bmatrix} \in \mathbb{R}^{n_2 \times l}, \tag{47}$$

$$\boldsymbol{\Phi}_{\mathrm{n}}(t) := [\boldsymbol{\varphi}(t), \cdots, \boldsymbol{\varphi}(t-l+1)] \in \mathbb{R}^{(n_c+n_d) \times l}. \tag{48}$$

Define two criterion functions:

$$J_3(\boldsymbol{a}) := \frac{1}{2}\|\boldsymbol{Y}(t) - \boldsymbol{\Phi}_{\mathrm{n}}^{\mathrm{T}}(t)\boldsymbol{\rho} - \boldsymbol{\Phi}_1^{\mathrm{T}}(t)\boldsymbol{a}\|^2, \tag{49}$$

$$J_4(\boldsymbol{\vartheta}) := \frac{1}{2}\|\boldsymbol{Y}(t) - \boldsymbol{\Psi}^{\mathrm{T}}(t)\boldsymbol{\vartheta}\|^2. \tag{50}$$

Let $k = 1, 2, 3, \cdots$ be an iterative variable, and $\hat{\boldsymbol{a}}_k(t)$, $\hat{\boldsymbol{\vartheta}}_k(t)$ be the iterative estimates of $\boldsymbol{a}$ and $\boldsymbol{\vartheta}$. For the optimization problems in (49)–(50), minimizing $J_3(\boldsymbol{a})$ and $J_4(\boldsymbol{\vartheta})$ and using the negative gradient search lead to the iterative estimates of $\boldsymbol{a}$ and $\boldsymbol{\vartheta}$:

$$\hat{\boldsymbol{a}}_k(t) = \hat{\boldsymbol{a}}_{k-1}(t) - \mu_{1,k}(t)\mathrm{grad}[J_3(\hat{\boldsymbol{a}}_{k-1}(t))]$$
$$= \hat{\boldsymbol{a}}_{k-1}(t) + \mu_{1,k}(t)\boldsymbol{\Phi}_1(t)$$
$$\times [\boldsymbol{Y}(t) - \boldsymbol{\Phi}_{\mathrm{n}}^{\mathrm{T}}(t)\hat{\boldsymbol{\rho}}_{k-1}(t) - \boldsymbol{\Phi}_1^{\mathrm{T}}(t)\hat{\boldsymbol{a}}_{k-1}(t)], \tag{51}$$

$$\hat{\boldsymbol{\vartheta}}_k(t) = \hat{\boldsymbol{\vartheta}}_{k-1}(t) - \mu_{2,k}(t)\mathrm{grad}[J_4(\hat{\boldsymbol{\vartheta}}_{k-1}(t))]$$
$$= \hat{\boldsymbol{\vartheta}}_{k-1}(t) + \mu_{2,k}(t)\boldsymbol{\Psi}_1(t)$$
$$\times [\boldsymbol{Y}(t) - \boldsymbol{\Psi}^{\mathrm{T}}(t)\hat{\boldsymbol{\vartheta}}_{k-1}(t)], \tag{52}$$

where $\mu_{1,k}(t) > 0$ and $\mu_{2,k}(t) > 0$ are the iterative step-sizes i.e., the convergence factors to be discussed later. Because Equations (51)–(52) contain the unknown $\boldsymbol{a}$, $\boldsymbol{b}$, $w(t-i)$ and $v(t-i)$, the approach here is based on the iterative principle and the hierarchical technique. Define the stacked information estimation $\hat{\boldsymbol{\varphi}}_k(t)$ and the stacked information estimation matrices $\hat{\boldsymbol{\Phi}}_{1,k}(t)$, $\hat{\boldsymbol{\Phi}}_{\mathrm{n},k}(t)$ and $\hat{\boldsymbol{\Psi}}_k(t)$ as

$$\hat{\boldsymbol{\varphi}}_k(t) := [-\hat{w}_{k-1}(t-1), -\hat{w}_{k-1}(t-2), \cdots, -\hat{w}_{k-1}(t-n_c),$$
$$\hat{v}_{k-1}(t-1), \hat{v}_{k-1}(t-2), \cdots, \hat{v}_{k-1}(t-n_d)]^{\mathrm{T}},$$

$$\hat{\boldsymbol{\Phi}}_{1,k}(t) := [\hat{\boldsymbol{\phi}}_{1,k}(t), \cdots, \hat{\boldsymbol{\phi}}_{1,k}(t-l+1)]$$
$$= [\boldsymbol{F}(t)\hat{\boldsymbol{b}}_{k-1}(t), \cdots, \boldsymbol{F}(t-l+1)\hat{\boldsymbol{b}}_{k-1}(t)],$$

$$\hat{\boldsymbol{\Phi}}_{\mathrm{n},k}(t) := [\hat{\boldsymbol{\varphi}}_k(t), \hat{\boldsymbol{\varphi}}_k(t-1), \cdots, \hat{\boldsymbol{\varphi}}_k(t-l+1)],$$

$$\hat{\boldsymbol{\Psi}}_k(t) := [\hat{\boldsymbol{\psi}}(t), \cdots, \hat{\boldsymbol{\psi}}(t-l+1)]$$
$$= \begin{bmatrix} \boldsymbol{F}^{\mathrm{T}}(t)\hat{\boldsymbol{a}}_{k-1}(t) & \cdots & \boldsymbol{F}^{\mathrm{T}}(t-l+1)\hat{\boldsymbol{a}}_{k-1}(t) \\ \hat{\boldsymbol{\varphi}}_k(t) & \cdots & \hat{\boldsymbol{\varphi}}_k(t-l+1) \end{bmatrix}.$$

From (8), we have $w(t-i) = y(t-i) - \boldsymbol{\phi}_1^{\mathrm{T}}(t-i)\boldsymbol{a}$. Replacing $\boldsymbol{\phi}_1(t-i)$ and $\boldsymbol{a}$ with $\hat{\boldsymbol{\phi}}_{1,k}(t-i)$ and $\hat{\boldsymbol{a}}_k(t)$, the estimate $\hat{w}_k(t-$

i) of $w(t-i)$ can be computed by

$$\hat{w}_k(t-i) = y(t-i) - \hat{\phi}_{1,k}^{\mathrm{T}}(t-i)\hat{a}_k(t).$$

Similarly, Equation (9) can be equivalently written as $v(t-i) = w(t-i) - \boldsymbol{\varphi}^{\mathrm{T}}(t-i)\boldsymbol{\rho}$. Replacing $\boldsymbol{\varphi}(t-i)$ and $\boldsymbol{\rho}$ with $\hat{\boldsymbol{\varphi}}_k(t-i)$ and $\hat{\boldsymbol{\rho}}_k(t)$ gives the estimate $\hat{v}_k(t-i)$ of $v(t-i)$ as

$$\hat{v}_k(t-i) = \hat{w}_k(t-i) - \hat{\boldsymbol{\varphi}}_k^{\mathrm{T}}(t-i)\hat{\boldsymbol{\rho}}_k(t).$$

Replacing $\boldsymbol{\Phi}_1(t)$ and $\boldsymbol{\Phi}_{\mathrm{n}}(t)$ in (51) with their estimates $\hat{\boldsymbol{\Phi}}_{1,k}(t)$ and $\hat{\boldsymbol{\Phi}}_{\mathrm{n},k}(t)$, and replacing $\boldsymbol{\Psi}(t)$ in (52) with its estimate $\hat{\boldsymbol{\Psi}}_k(t)$, we have

$$\begin{aligned}
\hat{a}_k(t) &= \hat{a}_{k-1}(t) + \mu_{1,k}(t)\hat{\boldsymbol{\Phi}}_{1,k}(t) \\
&\quad \times [\boldsymbol{Y}(t) - \hat{\boldsymbol{\Phi}}_{\mathrm{n},k}^{\mathrm{T}}(t)\hat{\boldsymbol{\rho}}_{k-1}(t) - \hat{\boldsymbol{\Phi}}_{1,k}^{\mathrm{T}}(t)\hat{a}_{k-1}(t)] \\
&= [\boldsymbol{I}_m - \mu_{1,k}(t)\hat{\boldsymbol{\Phi}}_{1,k}(t)\hat{\boldsymbol{\Phi}}_{1,k}^{\mathrm{T}}(t)]\hat{a}_{k-1}(t) \\
&\quad + \mu_{1,k}(t)\hat{\boldsymbol{\Phi}}_{1,k}(t)[\boldsymbol{Y}(t) - \hat{\boldsymbol{\Phi}}_{\mathrm{n},k}^{\mathrm{T}}(t)\hat{\boldsymbol{\rho}}_{k-1}(t)], \quad (53)
\end{aligned}$$

$$\begin{aligned}
\hat{\boldsymbol{\vartheta}}_k(t) &= \hat{\boldsymbol{\vartheta}}_{k-1}(t) + \mu_{2,k}(t)\hat{\boldsymbol{\Psi}}_k(t) \\
&\quad \times [\boldsymbol{Y}(t) - \hat{\boldsymbol{\Psi}}_k^{\mathrm{T}}(t)\hat{\boldsymbol{\vartheta}}_{k-1}(t)] \\
&= [\boldsymbol{I}_{n_1} - \mu_{2,k}(t)\hat{\boldsymbol{\Psi}}_k(t)\hat{\boldsymbol{\Psi}}_k^{\mathrm{T}}(t)]\hat{\boldsymbol{\vartheta}}_{k-1}(t) \\
&\quad + \mu_{2,k}(t)\hat{\boldsymbol{\Psi}}_k(t)\boldsymbol{Y}(t). \quad (54)
\end{aligned}$$

The above equations can be seen as discrete-time systems. In order to guarantee the convergence of $\hat{a}_k(t)$ and $\hat{\boldsymbol{\vartheta}}_k(t)$, the symmetric matrices $[\boldsymbol{I}_m - \mu_{1,k}(t)\hat{\boldsymbol{\Phi}}_{1,k}(t)\hat{\boldsymbol{\Phi}}_{1,k}^{\mathrm{T}}(t)]$ and $[\boldsymbol{I}_{n_1} - \mu_{2,k}(t)\hat{\boldsymbol{\Psi}}_k(t)\hat{\boldsymbol{\Psi}}_k^{\mathrm{T}}(t)]$ have all eigenvalues inside the unit circle. One conservative way is to choose of $\mu_{1,k}(t)$ and $\mu_{2,k}(t)$ to satisfy

$$\begin{aligned}
\mu_{1,k}(t) &\leq 2\lambda_{\max}^{-1}[\hat{\boldsymbol{\Phi}}_{1,k}(t)\hat{\boldsymbol{\Phi}}_{1,k}^{\mathrm{T}}(t)], \\
\mu_{2,k}(t) &\leq 2\lambda_{\max}^{-1}[\hat{\boldsymbol{\Psi}}_k(t)\hat{\boldsymbol{\Psi}}_k^{\mathrm{T}}(t)].
\end{aligned}$$

Computing the maximum eigenvalue of a square matrix requires heavier computational costs than computing its trace. The maximum eigenvalue of a positive definite matrix is always smaller than its trace. In order to reduce computational burden, an alternative way of $\mu_{1,k}(t)$ and $\mu_{2,k}(t)$ is to take

$$\begin{aligned}
\mu_{1,k}(t) &\leq 2\|\hat{\boldsymbol{\Phi}}_{1,k}(t)\|^{-2}, \\
\mu_{2,k}(t) &\leq 2\|\hat{\boldsymbol{\Psi}}_k(t)\|^{-2}.
\end{aligned}$$

Thus, the decomposition based generalized extended gradient iterative (D-GEGI) algorithm for computing the parameter estimation vectors $\hat{a}_k(t)$ and $\hat{\boldsymbol{\vartheta}}_k(t)$ is summarized as

$$\begin{aligned}
\bar{a}_k(t) &= \hat{a}_{k-1}(t) + \mu_{1,k}(t)\hat{\boldsymbol{\Phi}}_{1,k}(t)[\boldsymbol{Y}(t) \\
&\quad - \hat{\boldsymbol{\Phi}}_{\mathrm{n},k}^{\mathrm{T}}(t)\hat{\boldsymbol{\rho}}_{k-1}(t) - \hat{\boldsymbol{\Phi}}_{1,k}^{\mathrm{T}}(t)\hat{a}_{k-1}(t)], \quad (55)
\end{aligned}$$

$$\begin{aligned}
\hat{\boldsymbol{\vartheta}}_k(t) &= \hat{\boldsymbol{\vartheta}}_{k-1}(t) + \mu_{2,k}(t)\hat{\boldsymbol{\Psi}}_k(t) \\
&\quad \times [\boldsymbol{Y}(t) - \hat{\boldsymbol{\Psi}}_k^{\mathrm{T}}(t)\hat{\boldsymbol{\vartheta}}_{k-1}(t)], \quad (56)
\end{aligned}$$

$$\boldsymbol{Y}(t) = [y(t), y(t-1), \cdots, y(t-l+1)]^{\mathrm{T}}, \quad (57)$$

$$\hat{\boldsymbol{\varphi}}_k(t) = [-\hat{w}_{k-1}(t-1), \cdots, -\hat{w}_{k-1}(t-n_c), \quad (58)$$

$$\hat{v}_{k-1}(t-1), \cdots, \hat{v}_{k-1}(t-n_d)]^{\mathrm{T}}, \quad (59)$$

$$\begin{aligned}
\hat{\boldsymbol{\Phi}}_{1,k}(t) &= [\hat{\phi}_{1,k}(t), \cdots, \hat{\phi}_{1,k}(t-l+1)] \\
&= [\boldsymbol{F}(t)\hat{b}_{k-1}(t), \cdots, \boldsymbol{F}(t-l+1)\hat{b}_{k-1}(t)], \quad (60)
\end{aligned}$$

$$\hat{\boldsymbol{\Phi}}_{\mathrm{n},k}(t) = [\hat{\boldsymbol{\varphi}}_k(t), \hat{\boldsymbol{\varphi}}_k(t-1), \cdots, \hat{\boldsymbol{\varphi}}_k(t-l+1)], \quad (61)$$

$$\hat{\boldsymbol{\Psi}}_k(t) = [\hat{\psi}_k(t), \hat{\psi}_k(t-1), \cdots, \hat{\psi}_k(t-l+1)], \quad (62)$$

$$\hat{w}_k(t-i) = y(t-i) - \hat{\phi}_{1,k}^{\mathrm{T}}(t-i)\hat{a}_k(t), \quad (63)$$

$$\hat{v}_k(t-i) = \hat{w}_k(t-i) - \hat{\boldsymbol{\varphi}}_k^{\mathrm{T}}(t-i)\hat{\boldsymbol{\rho}}_k(t), \quad (64)$$

$$\mu_{1,k}(t) \leq 2\|\hat{\boldsymbol{\Phi}}_{1,k}(L)\|^{-2}, \quad (65)$$

$$\mu_{2,k}(t) \leq 2\|\hat{\boldsymbol{\Psi}}_k(L)\|^{-2}, \quad (66)$$

$$\begin{aligned}
\hat{\boldsymbol{\vartheta}}_k(t) &= [\hat{b}_{1,k}(t), \cdots, \hat{b}_{n,k}(t), \hat{c}_{1,k}(t), \cdots, \hat{c}_{n_c,k}(t), \\
&\quad \hat{d}_{1,k}(t), \cdots, \hat{d}_{n_d,k}(t)]^{\mathrm{T}}, \quad (67)
\end{aligned}$$

$$\hat{a}_k(t) = \mathrm{sgn}[\bar{a}_{1,k}(t)]\frac{\bar{a}_k(t)}{\|\bar{a}_k(t)\|}. \quad (68)$$

The D-GEGI algorithm can improve the parameter estimation accuracy by taking full advantage of all the collected measurements. Although the algorithm requires large computational efforts, this increased computation is still tolerable and affordable for modern computers. Here, if we set $l = L$ and $t = L$ in (45)–(48) (L: the data length), then

$$\boldsymbol{Y}(L) := [y(L), y(L-1), \cdots, y(1)]^{\mathrm{T}} \in \mathbb{R}^L, \quad (69)$$

$$\begin{aligned}
\boldsymbol{\Phi}_1(L) &:= [\phi_1(L), \phi_1(L-1), \cdots, \phi_1(1)] \\
&= [\boldsymbol{F}(L)\boldsymbol{b}, \boldsymbol{F}(L-1)\boldsymbol{b}, \cdots, \boldsymbol{F}(1)\boldsymbol{b}] \in \mathbb{R}^{m \times L}, \quad (70)
\end{aligned}$$

$$\begin{aligned}
\boldsymbol{\Psi}(L) &:= [\psi(L), \cdots, \psi(1)] \\
&= \begin{bmatrix} \boldsymbol{F}^{\mathrm{T}}(L)\boldsymbol{a} & \cdots & \boldsymbol{F}^{\mathrm{T}}(1)\boldsymbol{a} \\ \boldsymbol{\varphi}(L) & \cdots & \boldsymbol{\varphi}(1) \end{bmatrix} \in \mathbb{R}^{n_1 \times L}, \quad (71)
\end{aligned}$$

$$\boldsymbol{\Phi}_{\mathrm{n}}(L) := [\boldsymbol{\varphi}(L), \boldsymbol{\varphi}(L-1), \cdots, \boldsymbol{\varphi}(1)] \in \mathbb{R}^{(n_c+n_d) \times L}. \quad (72)$$

They contain all the measured data available $\{u(i), y(i) : i = 1, \cdots, L\}$. Define two criterion functions:

$$J_5(\boldsymbol{a}) := \frac{1}{2}\|\boldsymbol{Y}(L) - \boldsymbol{\Phi}_{\mathrm{n}}^{\mathrm{T}}(L)\boldsymbol{\rho} - \boldsymbol{\Phi}_1^{\mathrm{T}}(L)\boldsymbol{a}\|^2,$$

$$J_6(\boldsymbol{\vartheta}) := \frac{1}{2}\|\boldsymbol{Y}(L) - \boldsymbol{\Psi}^{\mathrm{T}}(L)\boldsymbol{\vartheta}\|^2.$$

Similarly, minimizing the quadratic criterion functions $J_5(\boldsymbol{a})$ and $J_6(\boldsymbol{\vartheta})$ yields the following D-GEGI algorithm:

$$\begin{aligned}
\bar{a}_k &= \hat{a}_{k-1} + \mu_{1,k}\hat{\boldsymbol{\Phi}}_{1,k}(L) \\
&\quad \times [\boldsymbol{Y}(L) - \hat{\boldsymbol{\Phi}}_{\mathrm{n},k}^{\mathrm{T}}(L)\hat{\boldsymbol{\rho}}_{k-1} - \hat{\boldsymbol{\Phi}}_{1,k}^{\mathrm{T}}(L)\hat{a}_{k-1}], \quad (73)
\end{aligned}$$

$$\hat{\boldsymbol{\vartheta}}_k = \hat{\boldsymbol{\vartheta}}_{k-1} + \mu_{2,k}\hat{\boldsymbol{\Psi}}_k(L)[\boldsymbol{Y}(L) - \hat{\boldsymbol{\Psi}}_k^{\mathrm{T}}(L)\hat{\boldsymbol{\vartheta}}_{k-1}], \quad (74)$$

$$\boldsymbol{Y}(L) = [y(L), y(L-1), \cdots, y(1)]^{\mathrm{T}}, \quad (75)$$

$$\hat{\boldsymbol{\varphi}}_k(t) = [-\hat{w}_{k-1}(t-1), \cdots, -\hat{w}_{k-1}(t-n_c), \quad (76)$$

$$\hat{v}_{k-1}(t-1), \cdots, \hat{v}_{k-1}(t-n_d)]^{\mathrm{T}}, \quad (77)$$

$$\begin{aligned}
\hat{\boldsymbol{\Phi}}_{1,k}(L) &= [\hat{\phi}_{1,k}(L), \hat{\phi}_{1,k}(L-1), \cdots, \hat{\phi}_{1,k}(1)] \\
&= [\boldsymbol{F}(L)\hat{b}_{k-1}, \boldsymbol{F}(L-1)\hat{b}_{k-1}, \cdots, \boldsymbol{F}(1)\hat{b}_{k-1}], \quad (78)
\end{aligned}$$

$$\hat{\boldsymbol{\Phi}}_{\mathrm{n},k}(L) = [\hat{\boldsymbol{\varphi}}_k(L), \hat{\boldsymbol{\varphi}}_k(L-1), \cdots, \hat{\boldsymbol{\varphi}}_k(1)], \quad (79)$$

$$\begin{aligned}
\hat{\boldsymbol{\Psi}}_k(L) &= [\hat{\psi}_k(L), \cdots, \hat{\psi}_k(1)] \\
&= \begin{bmatrix} \boldsymbol{F}^{\mathrm{T}}(L)\hat{a}_{k-1} & \cdots & \boldsymbol{F}^{\mathrm{T}}(1)\hat{a}_{k-1} \\ \hat{\boldsymbol{\varphi}}_k(L) & \cdots & \hat{\boldsymbol{\varphi}}_k(1) \end{bmatrix}, \quad (80)
\end{aligned}$$

$$\hat{w}_k(t) = y(t) - \hat{\phi}_{1,k}^{\mathrm{T}}(t)\hat{a}_k, \quad (81)$$

$$\hat{v}_k(t) = \hat{w}_k(t) - \hat{\boldsymbol{\varphi}}_k^{\mathrm{T}}(t)\hat{\boldsymbol{\rho}}_k, \quad (82)$$

$$\mu_{1,k} \leq 2\|\hat{\boldsymbol{\Phi}}_{1,k}(L)\|^{-2}, \quad (83)$$

$$\mu_{2,k} \leq 2\|\hat{\boldsymbol{\Psi}}_k(L)\|^{-2}, \quad (84)$$

$$\begin{aligned}
\hat{\boldsymbol{\vartheta}}_k &= [\hat{b}_{1,k}, \cdots, \hat{b}_{n,k}, \hat{c}_{1,k}, \cdots, \hat{c}_{n_c,k}, \\
&\quad \hat{d}_{1,k}, \cdots, \hat{d}_{n_d,k}]^{\mathrm{T}}, \quad (85)
\end{aligned}$$

$$\hat{\boldsymbol{a}}_k = \text{sgn}[\bar{a}_{1,k}] \frac{\bar{\boldsymbol{a}}_k}{\|\bar{\boldsymbol{a}}_k\|}. \tag{86}$$

To summarize, we list the steps involved in the D-GEGI algorithm to estimate $\boldsymbol{a}$ and $\boldsymbol{\vartheta}$ as k increases:

1. Set the data length L and collect the input-output data $u(t)$ and $y(t)$, $t = 1, 2, \cdots, L$. Form $\boldsymbol{F}(t)$ by (2) and $\boldsymbol{Y}(L)$ by (75).
2. Let $k = 1$, and give a small positive number ε and the nonlinear basis function $f_i(*)$. Set the initial values: $\hat{\boldsymbol{a}}_0 = \boldsymbol{1}_m/\sqrt{m}$, $\hat{\boldsymbol{\vartheta}}_0 = \boldsymbol{1}_{n+n_c+n_d}/p_0$, $p_0 = 10^6$, $\hat{w}_0(t)$ and $\hat{v}_0(t)$ are two random numbers.
3. Form $\hat{\boldsymbol{\varphi}}_k(t)$ using (76), $\hat{\boldsymbol{\Phi}}_{n,k}(t)$ using (79), $\hat{\boldsymbol{\Phi}}_{1,k}(t)$ using (78) and $\hat{\boldsymbol{\Psi}}_k(t)$ using (80).
4. Select a large $\mu_{1,k}$ according to (83) and update the parameter estimation vector $\hat{\boldsymbol{a}}_k$ using (73) and normalize $\hat{\boldsymbol{a}}_k$ using (86).
5. Select a large $\mu_{2,k}$ satisfying (84) and update the parameter estimation vector $\hat{\boldsymbol{\vartheta}}_k$ using (74).
6. Compute $\hat{w}_k(t)$ and $\hat{v}_k(t)$ using (81)–(82).
7. If $\|\hat{\boldsymbol{a}}_k - \hat{\boldsymbol{a}}_{k-1}\| + \|\hat{\boldsymbol{\vartheta}}_k - \hat{\boldsymbol{\vartheta}}_{k-1}\| > \varepsilon$, increase k by 1 and go to Step 3, otherwise, obtain the parameter estimation vectors $\hat{\boldsymbol{a}}_k$ and $\hat{\boldsymbol{\vartheta}}_k$.

Compared with the D-GESG and the D-MI-GESG algorithms, the D-GEGI algorithm refreshes the parameter estimates $\hat{\boldsymbol{a}}_k$ and $\hat{\boldsymbol{\vartheta}}_k$ by using a fixed data batch with a finite length L and is able to take sufficient advantage of all the measured data at each iteration so that the parameter estimation accuracy can be greatly improved.

The computational complexity of the algorithm is evaluated by the number of multiplications and additions. One multiplication or one addition is called a flop, which means floating-point operation. The sum of the flops implies the computational complexity of the algorithm. The computational efficiency of the D-GESG algorithm, the D-MI-GESG algorithm and the D-GEGI algorithm are given in Tables 1–3 ($n_1 = m + n_c + n_d$, $n_2 = n + n_c + n_d$, $n_0 = m + n + n_c + n_d$).

Obviously, the D-GESG algorithm requires small computational cost than the D-MI-GESG algorithm at each recursive step. In Section 5, we will give the comparison of the computational load for the D-MI-GESG algorithm and the D-GEGI algorithm.

5 Example

Consider the Hammerstein finite impulse response system shown in Figure 1, the nonlinear function $g(*)$ is a linear combination of known basis functions $\boldsymbol{g}(u(t)) := [g_1(u(t)), g_2(u(t)), \cdots, g_m(u(t))]^{\mathrm{T}} \in \mathbb{R}^m$ and coefficients $(a_1, a_2, \cdots, a_m)$:

$$\bar{u}(t) = g(u(t)) = a_1 g_1(u(t)) + a_2 g_2(u(t)) + \cdots + a_m g_m(u(t)),$$

the linear dynamic block is a finite impulse response with a coefficient $(b_1, b_2, \cdots, b_n)$:

$$B(z) = b_1 z^{-1} + b_2 z^{-2} + \cdots + b_n z^{-n}.$$

Define the information matrix

$$\boldsymbol{G}(t) := [\boldsymbol{g}(u(t-1)), \boldsymbol{g}(u(t-2)), \cdots, \boldsymbol{g}(u(t-n))] \in \mathbb{R}^{m \times n}.$$

From Figure 1, we have

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

$$= b_1\bar{u}(t-1) + \cdots + b_n\bar{u}(t-n) + \frac{D(z)}{C(z)}v(t)$$

$$= [\boldsymbol{a}^{\mathrm{T}}\boldsymbol{g}(u(t-1)), \cdots, \boldsymbol{a}^{\mathrm{T}}\boldsymbol{g}(u(t-n))]\boldsymbol{b} + \frac{D(z)}{C(z)}v(t)$$

$$= \boldsymbol{a}^{\mathrm{T}}\boldsymbol{G}(t)\boldsymbol{b} + \frac{D(z)}{C(z)}v(t),$$

$$\boldsymbol{G}(t) = \left[\begin{array}{cccc} u(t-1) & u(t-2) & u(t-3) & u(t-4) \\ u^2(t-1) & u^2(t-2) & u^2(t-3) & u^2(t-4) \end{array} \right],$$

$$C(z) = 1 + c_1 z^{-1} = 1 + 0.20 z^{-1},$$

$$D(z) = 1 + d_1 z^{-1} = 1 + 0.23 z^{-1},$$

$$\boldsymbol{a} = [a_1, a_2]^{\mathrm{T}} = [0.30, \sqrt{0.91}]^{\mathrm{T}},$$

$$\boldsymbol{b} = [b_1, b_2, b_3, b_4]^{\mathrm{T}} = [0.54, 1.25, 1.5, 1.5]^{\mathrm{T}},$$

$$\boldsymbol{\theta} = [0.30, \sqrt{0.91}, 0.54, 1.25, 1.5, 1.5, 0.20, 0.23]^{\mathrm{T}}.$$

Thus, the Hammerstein finite impulse response system in Figure 1 is transformed into a bilinear-in-parameter model like in (4).
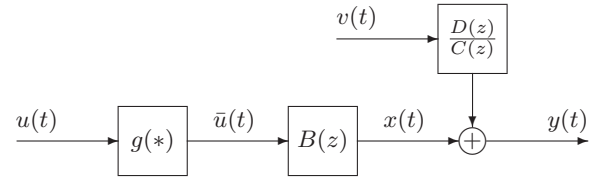

Fig. 1: The Hammerstein nonlinear finite impulse response system

In simulation, the input $\{u(t)\}$ is taken as an uncorrelated stochastic signal sequence with zero mean and unit variance, $\{v(t)\}$ as a white noise sequence with zero mean and variance σ^2.

Apply the D-MI-GESG algorithm in (28)–(44) and the D-GEGI algorithm in (73)–(86) to estimate the parameters of this system with $L = 3000$. The D-MI-GESG parameter estimates and their errors are shown in Table 4 with $p = 1, 2$ and 3, the parameter estimation errors $\delta_t := \|\hat{\boldsymbol{\theta}}(t) - \boldsymbol{\theta}\|/\|\boldsymbol{\theta}\|$ versus t are shown in Figure 2. When $p = 3$, the parameter estimation errors δ_t versus t with different $\sigma^2 = 0.20^2, 2.00^2, 4.00^2$ are given in Figures 3. For comparison, we use the D-GEGI algorithm to estimate the unknown parameters, Table 5 gives the parameter estimates and their errors $\delta_k := \|\hat{\boldsymbol{\theta}}_k - \boldsymbol{\theta}\|/\|\boldsymbol{\theta}\|$ under different noise levels and Figure 4 shows the parameter estimation errors δ_k versus k.
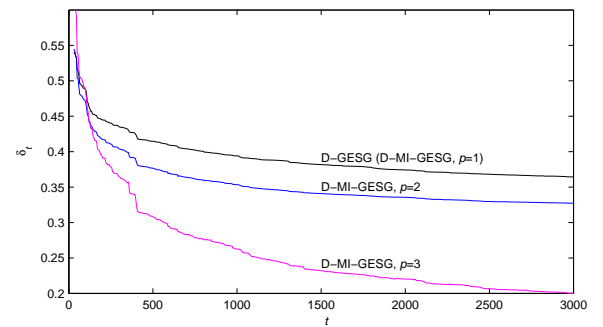

Fig. 2: The D-MI-GESG estimation errors δ_t versus t with $\sigma^2 = 0.20^2$

From Tables 1–3, we can see that the computational efficiencies of the D-GESG, D-MI-GESG and D-GEGI algorithms are N_1 flops, N_2 flops and N_3 flops at each step. In this example, the total flops of the D-GESG, D-MI-GESG and D-GEGI algorithms in running process are $\Sigma_1 := N_1 t = 282000$, $\Sigma_2 := N_2 t = 726000$ and $\Sigma_3 := N_3 k = 30000800$.

From the results of this simulation example, the following conclusions can be drawn.

Table 1 The computational efficiency of the D-GESG algorithm

Variables	Expressions	Multiplications	Additions
$\hat{\boldsymbol{a}}(t)$	$\hat{\boldsymbol{a}}(t) = \hat{\boldsymbol{a}}(t-1) + \frac{\hat{\boldsymbol{\phi}}_1(t)}{r_1(t)} e_1(t) \in \mathbb{R}^m$	$2m$	m
	$e_1(t) = y(t) - \hat{\boldsymbol{\phi}}_1^{\mathrm{T}}(t)\hat{\boldsymbol{a}}(t-1) - \hat{\boldsymbol{\varphi}}^{\mathrm{T}}(t)\hat{\boldsymbol{\rho}}(t-1) \in \mathbb{R}$	$m + n_c + n_d$	$m + n_c + n_d$
	$r_1(t) = r_1(t-1) + \|\hat{\boldsymbol{\phi}}_1(t)\|^2 \in \mathbb{R}$	m	m
	$\hat{\boldsymbol{\phi}}_1(t) = \boldsymbol{F}(t)\hat{\boldsymbol{b}}(t-1) \in \mathbb{R}^m$	mn	$m(n-1)$
$\hat{\boldsymbol{\vartheta}}(t)$	$\hat{\boldsymbol{\vartheta}} = \hat{\boldsymbol{\vartheta}}(t-1) + \frac{\hat{\boldsymbol{\psi}}(t)}{r_2(t)} e_2(t) \in \mathbb{R}^{n+n_c+n_d}$	$2(n + n_c + n_d)$	$n + n_c + n_d$
	$e_2(t) = y(t) - \hat{\boldsymbol{\psi}}^{\mathrm{T}}(t)\hat{\boldsymbol{\vartheta}}(t-1) \in \mathbb{R}$	$n + n_c + n_d$	$n + n_c + n_d$
	$r_2(t) = r_2(t-1) + \|\hat{\boldsymbol{\psi}}(t)\|^2 \in \mathbb{R}$	$n + n_c + n_d$	$n + n_c + n_d$
	$\hat{\boldsymbol{\phi}}_2(t) = \boldsymbol{F}^{\mathrm{T}}(t)\hat{\boldsymbol{a}}(t-1) \in \mathbb{R}^n$	mn	$n(m-1)$
$\hat{v}(t)$	$\hat{v}(t) = \hat{w}(t) - \hat{\boldsymbol{\varphi}}^{\mathrm{T}}(t)\hat{\boldsymbol{\rho}} \in \mathbb{R}$	$n_c + n_d$	$n_c + n_d$
$\hat{w}(t)$	$\hat{w}(t) = y(t) - \hat{\boldsymbol{\phi}}_1^{\mathrm{T}}(t)\hat{\boldsymbol{a}} \in \mathbb{R}$	m	m
Total flops		$N_1 := 6n_0 + 2n_1 + 4mn + 3(n_c + n_d)$	

Table 2 The computational efficiency of the D-MI-GESG algorithm

Variables	Expressions	Multiplications	Additions
$\hat{\boldsymbol{a}}(t)$	$\hat{\boldsymbol{a}}(t) = \hat{\boldsymbol{a}}(t-1) + \frac{\hat{\boldsymbol{\Phi}}_1(p,t)}{r_1(t)} \boldsymbol{E}_1(p,t) \in \mathbb{R}^m$	$m(p+1)$	mp
	$\boldsymbol{E}_1(p,t) = \boldsymbol{Y}(p,t) - \hat{\boldsymbol{\Phi}}_1^{\mathrm{T}}(p,t)\hat{\boldsymbol{a}}(t-1) - \hat{\boldsymbol{\Phi}}_{\mathrm{n}}^{\mathrm{T}}(p,t)\hat{\boldsymbol{\rho}}(t-1) \in \mathbb{R}^p$	$(m + n_c + n_d)p$	$(m + n_c + n_d)p$
	$r_1(t) = r_1(t-1) + \|\hat{\boldsymbol{\Phi}}_1(p,t)\|^2 \in \mathbb{R}$	mp	mp
	$\hat{\boldsymbol{\Phi}}_1(p,t) = [\boldsymbol{F}(t)\hat{\boldsymbol{b}}(t-1), \cdots, \boldsymbol{F}(t-p+1)\hat{\boldsymbol{b}}(t-1)] \in \mathbb{R}^{m \times p}$	mnp	$m(n-1)p$
$\hat{\boldsymbol{\vartheta}}(t)$	$\hat{\boldsymbol{\vartheta}} = \hat{\boldsymbol{\vartheta}}(t-1) + \frac{\hat{\boldsymbol{\Psi}}(p,t)}{r_2(t)} \boldsymbol{E}_2(p,t) \in \mathbb{R}^{n+n_c+n_d}$	$(n + n_c + n_d)(p+1)$	$(n + n_c + n_d)p$
	$\boldsymbol{E}_2(p,t) = \boldsymbol{Y}(p,t) - \hat{\boldsymbol{\Psi}}^{\mathrm{T}}(p,t)\hat{\boldsymbol{\vartheta}}(t-1) \in \mathbb{R}^p$	$(n + n_c + n_d)p$	$(n + n_c + n_d)p$
	$r_2(t) = r_2(t-1) + \|\hat{\boldsymbol{\Psi}}(p,t)\|^2 \in \mathbb{R}$	$(n + n_c + n_d)p$	$(n + n_c + n_d)p$
	$\hat{\boldsymbol{\Psi}}(p,t) = \begin{bmatrix} \boldsymbol{F}^{\mathrm{T}}(t)\hat{\boldsymbol{a}}(t-1) & \cdots & \boldsymbol{F}^{\mathrm{T}}(t-p+1)\hat{\boldsymbol{a}}(t-1) \\ \hat{\boldsymbol{\varphi}}(t) & \cdots & \hat{\boldsymbol{\varphi}}(t-p+1) \end{bmatrix} \in \mathbb{R}^{n_2 \times p}$	mnp	$n(m-1)p$
$\hat{v}(t)$	$\hat{v}(t) = \hat{w}(t) - \hat{\boldsymbol{\varphi}}^{\mathrm{T}}(t)\hat{\boldsymbol{\rho}} \in \mathbb{R}$	$(n_c + n_d)p$	$(n_c + n_d)p$
$\hat{w}(t)$	$\hat{w}(t) = y(t) - \hat{\boldsymbol{\phi}}_1^{\mathrm{T}}(t)\hat{\boldsymbol{a}} \in \mathbb{R}$	mp	mp
Total flops		$N_2 := 4n_1 p + 6n_2 p + 4mnp + 4mp + n_0 - np$	

Table 3 The computational efficiency of the D-GEGI algorithm

Variables	Expressions	Multiplications	Additions
$\hat{\boldsymbol{a}}_k$	$\hat{\boldsymbol{a}}_k = \hat{\boldsymbol{a}}_{k-1} + \mu_{1,k}\hat{\boldsymbol{\Phi}}_{1,k}(L)\boldsymbol{E}_{1,k} \in \mathbb{R}^m$	$mL + m$	mL
$\mu_{1,k}$	$\mu_{1,k} \leq 2\|\hat{\boldsymbol{\Phi}}_{1,k}(L)\|^{-2} \in \mathbb{R}$	$mL + 1$	$mL - 1$
$\boldsymbol{E}_{1,k}$	$\boldsymbol{E}_{1,k} := \boldsymbol{Y}(L) - \hat{\boldsymbol{\Phi}}_{n,k}^{\mathrm{T}}(L)\hat{\boldsymbol{\rho}}_{k-1} - \hat{\boldsymbol{\Phi}}_{1,k}^{\mathrm{T}}(L)\hat{\boldsymbol{a}}_{k-1} \in \mathbb{R}^L$	$(m + n_c + n_d)L$	$(m + n_c + n_d)L$
$\hat{\boldsymbol{\vartheta}}_k$	$\hat{\boldsymbol{\vartheta}}_k = \hat{\boldsymbol{\vartheta}}_{k-1} + \mu_{2,k}\hat{\boldsymbol{\Psi}}_k(L)\boldsymbol{E}_{2,k} \in \mathbb{R}^{n+n_c+n_d}$	$(n + n_c + n_d)L + n + n_c + n_d$	$(n + n_c + n_d)L$
$\mu_{2,k}$	$\mu_{2,k} \leq 2\|\hat{\boldsymbol{\Psi}}_k(L)\|^{-2} \in \mathbb{R}$	$(n + n_c + n_d)L + 1$	$(n + n_c + n_d)L - 1$
$\boldsymbol{E}_{2,k}$	$\boldsymbol{E}_{2,k} := \boldsymbol{Y}(L) - \hat{\boldsymbol{\Psi}}_k^{\mathrm{T}}(L)\hat{\boldsymbol{\vartheta}}_{k-1} \in \mathbb{R}^L$	$(n + n_c + n_d)L$	$(n + n_c + n_d)L$
$\hat{\boldsymbol{\phi}}_{1,k}(t)$	$\hat{\boldsymbol{\phi}}_{1,k}(t) = \boldsymbol{F}(t)\hat{\boldsymbol{b}}_{k-1} \in \mathbb{R}^m$	$2mnL$	$2(n-1)mL$
$\hat{\boldsymbol{\phi}}_{2,k}(t)$	$\hat{\boldsymbol{\phi}}_{2,k}(t) = \boldsymbol{F}^{\mathrm{T}}(t)\hat{\boldsymbol{a}}_{k-1} \in \mathbb{R}^n$	mnL	$(m-1)nL$
$\hat{v}_k(t)$	$\hat{v}_k(t) = \hat{w}_k(t) - \hat{\boldsymbol{\varphi}}_k^{\mathrm{T}}(t)\hat{\boldsymbol{\rho}}_k \in \mathbb{R}$	$(n_c + n_d)L$	$(n_c + n_d)L$
$\hat{w}_k(t)$	$\hat{w}_k(t) = y(t) - \hat{\boldsymbol{\phi}}_{1,k}^{\mathrm{T}}(t)\hat{\boldsymbol{a}}_k \in \mathbb{R}$	mL	mL
Total flops		$N_3 := (4n_1 + 6n_2 + 6mn + 2m - n)L + n_0$	

- The D-GESG algorithm has high estimation errors. Based on the multi-innovation theory, the D-MI-GESG algorithm make the parameter estimation errors become smaller when innovation length p increases. Moreover, the parameter estimation errors given by the D-MI-GESG algorithm become smaller as t increases.
- The parameter estimation accuracy of the proposed D-GEGI algorithm becomes higher with the increasing of iteration k and the parameter estimates are close to their true values.
- For the same data length and noise levels, it is obviously that parameter estimation accuracy of the D-GEGI algorithm is higher than that by the corresponding D-MI-GESG algorithm for bilinear-in-parameter systems. The results show that a lower noise level leads to a higher parameter estimation accuracy by both algorithms. The D-GEGI algorithm has the higher computational effort than the D-MI-GESG algorithm.
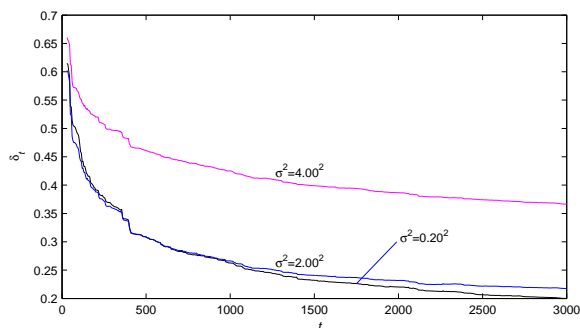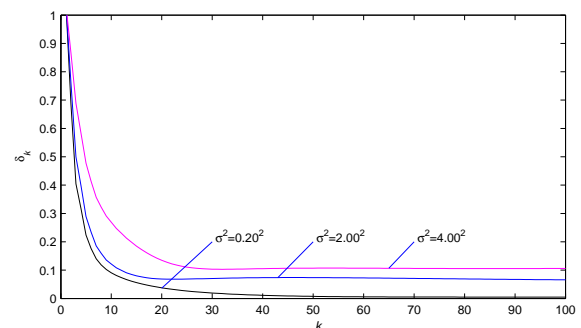
6 Conclusions

This paper considers the parameter estimation problems for bilinear-in-parameter systems by means of the negative gradient search. Based on the the model decomposition technique, we derive the D-GESG algorithm, the D-MI-GESG algorithm and the D-GEGI algorithm. The D-GEGI algorithm can generate more accurate

Table 4 The D-MI-GESG estimates and errors ($\sigma^2 = 0.20^2$)

p	t	a_1	a_2	b_1	b_2	b_3	b_4	c_1	d_1	$\delta_t(\%)$
1 (D-GESG)	100	0.08767	0.99615	0.52315	0.56623	1.65401	0.72225	-0.56227	0.01155	48.78112
	200	0.10277	0.99471	0.52408	0.65344	1.61509	0.81650	-0.50942	-0.07412	44.47296
	500	0.12137	0.99261	0.55131	0.72590	1.60439	0.89792	-0.49487	-0.11099	41.47031
	1000	0.12539	0.99211	0.56476	0.77897	1.59724	0.95942	-0.48259	-0.13627	39.40240
	2000	0.13437	0.99093	0.57783	0.83560	1.59293	1.02212	-0.47389	-0.15697	37.42139
	3000	0.13943	0.99023	0.58384	0.86443	1.58762	1.05408	-0.46778	-0.16765	36.42892
2	100	0.32342	0.94626	0.47977	0.94839	2.41918	0.84035	0.31580	-0.28117	47.06989
	200	0.32637	0.94524	0.42969	0.97905	2.29287	0.97741	0.35584	-0.30189	41.70560
	500	0.30556	0.95217	0.43551	1.01274	2.17058	1.08537	0.41366	-0.33790	37.67442
	1000	0.29945	0.95411	0.42411	1.01616	2.07946	1.15818	0.45340	-0.35728	35.33572
	2000	0.29188	0.95645	0.42359	1.02837	2.00459	1.22956	0.49189	-0.37220	33.55210
	3000	0.28994	0.95704	0.42007	1.03286	1.96435	1.26480	0.51319	-0.37559	32.72269
3	100	0.35312	0.93558	0.41925	0.90247	2.47060	0.87967	0.17868	-0.31087	48.59226
	200	0.32311	0.94636	0.35763	0.88653	2.23649	1.08311	0.22184	-0.27721	39.12418
	500	0.28842	0.95750	0.39645	0.90394	2.02294	1.24272	0.28025	-0.24232	30.87087
	1000	0.26929	0.96306	0.39055	0.87510	1.86225	1.33737	0.30436	-0.19563	26.26261
	2000	0.25766	0.96623	0.40118	0.87291	1.74563	1.42816	0.29937	-0.12015	22.02908
	3000	0.25311	0.96744	0.40424	0.86685	1.68222	1.46337	0.28337	-0.06714	19.98415
True values		0.30000	0.95394	0.54000	1.25000	1.50000	1.5000	0.20000	0.23000	

Table 5 The D-GEGI estimates and errors

σ^2	k	a_1	a_2	b_1	b_2	b_3	b_4	c_1	d_1	$\delta_k(\%)$
0.20^2	1	0.06399	0.99795	2.08017	2.63747	2.85895	2.85109	-0.12012	-0.00261	104.88295
	2	0.11902	0.99289	1.59114	2.15611	2.37300	2.36138	0.03553	0.25148	68.48643
	5	0.32034	0.94730	0.87561	1.49035	1.74024	1.74505	0.43594	0.03200	22.38965
	10	0.40419	0.91468	0.63494	1.32614	1.62037	1.63392	0.28494	0.23162	9.07650
	50	0.30104	0.95361	0.53403	1.24323	1.50986	1.51145	0.25965	0.25768	0.73961
	100	0.29620	0.95513	0.53733	1.24044	1.50176	1.50201	0.25721	0.26011	0.48144
2.00^2	1	0.06000	0.99820	2.06420	2.59927	2.87529	2.86543	-0.12202	-0.02169	104.52494
	2	0.11184	0.99373	1.67939	2.22121	2.49648	2.48243	-0.03191	0.22781	75.94932
	5	0.31126	0.95032	0.98593	1.56345	1.86171	1.85374	0.25581	-0.00939	28.91678
	10	0.40721	0.91334	0.68172	1.31271	1.64216	1.64042	0.11105	0.13963	12.17191
	20	0.32055	0.94723	0.55667	1.20384	1.52231	1.50766	0.09908	0.16467	6.86752
	100	0.24763	0.96885	0.54389	1.15762	1.43809	1.40469	0.16203	0.27270	6.57269
4.00^2	1	0.05558	0.99845	2.04646	2.55682	2.89345	2.88137	-0.12413	-0.04289	104.16434
	2	0.10346	0.99463	1.80822	2.32320	2.66128	2.64595	-0.12561	0.18221	86.57320
	5	0.32559	0.94551	1.26325	1.79136	2.13437	2.11111	0.05631	-0.01560	47.74720
	10	0.51978	0.85430	0.90771	1.46213	1.81849	1.79063	-0.03798	0.07431	26.90482
	20	0.41038	0.91192	0.67509	1.24668	1.60224	1.55800	-0.01410	0.11914	13.46115
	100	0.21365	0.97691	0.53874	1.08350	1.41699	1.34172	0.15949	0.35027	10.59357
True values		0.30000	0.95394	0.54000	1.25000	1.50000	1.5000	0.20000	0.23000	

Fig. 3: The D-MI-GESG estimation errors δ_t versus t with different σ^2

Fig. 4: The D-GEGI estimation errors δ_k versus k with different σ^2

parameter estimates than the D-GESG and the D-MI-GESG algorithms by exploiting all the measured input-output data at each iteration. The proposed algorithms have the advantage of not involving the over-parameterization of the parameters and decrease the computation load. The proposed approaches in the paper can combine other mathematical tools [45–47] and statistical strategies [48–51] to study

the performances of some parameter estimation algorithms and can be applied to other multivariable systems with different structures and disturbance noises and other literature [52–55].

7 Acknowledgments

8 References

1 Xu, L.: 'The parameter estimation algorithms based on the dynamical response measurement data', Adv. Mech. Eng., 2017, **9**, (11), pp. 1-12. doi: 10.1177/1687814017730003

2 Wang, Y.J., Ding, F., Xu, L.: 'Some new results of designing an IIR filter with colored noise for signal processing', Digit. Signal Process., 2018, **72**, pp. 44-58.

3 Lin, Y., Zhang, W.: 'Necessary/sufficient conditions for pareto optimum in cooperative difference game', Optim. Control Appl. Methods, 2018, **39**, (2), pp. 1043-1060.

4 Li, Y., Zhang, W.H., Liu, X.K.: 'H-index for discrete-time stochastic systems with Markovian jump and multiplicative noise', Automatica, 2018, **90**, pp. 286-293.

5 Zhang, W.H., Xue, L., Jiang, X.: 'Global stabilization for a class of stochastic nonlinear systems with SISS-like conditions and time delay', Int. J. Robust Nonlinear Control, 2018, **28**, (13), pp. 3909-3926.

6 Wang, Y., Zhang, H., Wei, S., *et al.*: 'Control performance assessment for ILC-controlled batch processes in two-dimensional system framework', IEEE Trans. Syst. Man Cybern. -Syst., 2018, **48**. doi: 10.1109/TSMC.2017.2672563

7 Wang, D.Q., Li, L.W., Ji, Y., *et al.*: 'Model recovery for Hammerstein systems using the auxiliary model based orthogonal matching pursuit method', Appl. Math. Model., 2018, **54**, pp. 537-550.

8 Li, J.H., Zheng, W.X., Gu, J.P., *et al.*: 'A recursive identification algorithm for Wiener nonlinear systems with linear state-space subsystem', Circuits Syst. Signal Process., 2018, **37**, (6), pp. 2374-2393.

9 Li, J.H., Li, X.: 'Particle swarm optimization iterative identification algorithm and gradient iterative identification algorithm for Wiener systems with colored noise', Complexity, 2018, Article ID 7353171, 1-8. https://doi.org/10.1155/2018/7353171

10 Allafi, W., Zajic, I., Uddin, K., *et al.*: 'Parameter estimation of the fractional-order Hammerstein-Wiener model using simplified refined instrumental variable fractional-order continuous time', IET Control Theory Appl., 2017, **11**, (15), pp. 2591-2598.

11 Wang, D.Q., Zhang, Z., Yuan J.Y.: 'Maximum likelihood estimation method for dual-rate Hammerstein systems', Int. J. Control Autom. Syst., 2017, **15**, (2), pp. 698-705.

12 Wang, Y.J., Ding, F.: 'A filtering based multi-innovation gradient estimation algorithm and performance analysis for nonlinear dynamical systems', IMA J. Appl. Math., 2017, **82**, (6), pp. 1171-1191.

13 Ding, F., Chen, H.B., Xu, L., *et al.*: 'A hierarchical least squares identification algorithm for Hammerstein nonlinear systems using the key term separation', J. Frankl. Inst., 2018, **355**, (8), pp. 3737-3752.

14 Bottegal, G., Castro-Garcia, R., Suykens, J.A.K.: 'A two-experiment approach to Wiener system identification', Automatica, 2018, **93**, pp. 282-289.

15 Li, L.W., Ren, X.M., Guo, F.M.: 'Modified multi-innovation stochastic gradient algorithm for Wiener-Hammerstein systems with backlash', J. Frankl. Inst., 2018, **355**, (9), pp. 4050-4075.

16 Cao, Y., Li, P., Zhang, Y.: 'Parallel processing algorithm for railway signal fault diagnosis data based on cloud computing', Futur. Gener. Comp. Syst., 2018, **88**, pp. 279-283.

17 Cao, Y., Ma, L.C., Xiao, S., *et al.*: 'Standard analysis for transfer delay in CTCS-3', Chin. J. Electron., 2017, **26**, (5), pp. 1057-1063.

18 Cao, Y., Wen, Y., Meng, X., *et al.*: 'Performance evaluation with improved receiver design for asynchronous coordinated multipoint transmissions', Chin. J. Electron., 2016, **25**, (2), pp. 372-378.

19 Zhang, Y.Z., Cao, Y., Wen, Y.H., *et al.*: 'Optimization of information interaction protocols in cooperative vehicle-infrastructure systems', Chin. J. Electron., 2018, **27**, (2), pp. 439-444.

20 Ding, F., Xu, L., *et al.*: 'Iterative parameter identification for pseudo-linear systems with ARMA noise using the filtering technique', IET Control Theory Appl., 2018, **12**, (7), pp. 892-899.

21 Mattsson, P., Zachariah, D., Stoica, P.: 'Recursive nonlinear-system identification using latent variables', 2018, Automatica, **93**, pp. 343-351.

22 Zhang, X., Ding, F., *et al.*: 'Recursive parameter identification of the dynamical models for bilinear state space systems', Nonlinear Dyn., 2017, **89**, (4), pp. 2415-2429.

23 Pan, J., Ma, H., Jiang, X., *et al.*: 'Adaptive gradient-based iterative algorithm for multivariate controlled autoregressive moving average systems using the data filtering technique', Complexity, 2018, Article ID 9598307. https://doi.org/10.1155/2018/9598307

24 Xu, L., Ding, F.: 'Parameter estimation for control systems based on impulse responses', Int. J. Control Autom. Syst., 2017, **15**, (6), pp. 2471-2479.

25 Ma, P., Ding, F., Zhu, Q.M.: 'Decomposition-based recursive least squares identification methods for multivariate pseudo linear systems using the multi-innovation', Int. J. Syst. Sci., 2018, **49**, (5), pp. 920-928.

26 Xu, L., Xiong, W.L., Alsaedi, A., *et al.*: 'Hierarchical parameter estimation for the frequency response based on the dynamical window data', Int. J. Control Autom. Syst., 2018, **16**, (4), pp. 1756-1764.

27 Li, M.H., Liu X.M.: 'Auxiliary model based least squares iterative algorithms for parameter estimation of bilinear systems using interval-varying measurements', IEEE Access, 2018, **6**, pp. 21518-21529.

28 Li, M.H., Liu, X.M.: 'The least squares based iterative algorithms for parameter estimation of a bilinear system with autoregressive noise using the data filtering technique', Signal Process., 2018, **147**, pp. 23-34.

29 Xu, L., Ding, F.: 'Iterative parameter estimation for signal models based on measured data', Circuits Syst. Signal Process., 2018, **37**, (7), pp. 3046-3069.

30 Paleologu, C., Benesty, J., Ciochină, S.: 'Adaptive filtering for the identification of bilinear forms', Digit. Signal Process., 2018, **75**, pp. 153-167.

31 Benesty, J., Paleologu, C., Ciochină, S.: 'On the identification of bilinear forms with the Wiener filter', IEEE Signal Process. Lett., 2017, **24**, (5), pp. 653-657.

32 Wang, D.Q., Mao, L., Ding, F.: 'Recasted models based hierarchical extended stochastic gradient method for MIMO nonlinear systems', IET Control Theory Appl., 2017, **11**, (4), pp. 476-485.

33 Wang, X.H., Ding, F., *et al.*: 'Convergence analysis of the hierarchical least squares algorithm for bilinear-in-parameter systems', Circuits Syst. Signal Process., 2016, **35**, (12), pp. 4307-4330.

34 Bai, E.W., Liu, Y.: 'Least squares solutions of bilinear equations', Syst. Control Lett., 2006, **55**, (6), pp. 466-472.

35 Filipovic, V.Z.: 'Consistency of the robust recursive Hammerstein model identification algorithm', J. Frankl. Inst., 2015, **352**, (5), pp. 1932-1945.

36 Zhang, X., Xu, L., *et al.*: 'Combined state and parameter estimation for a bilinear state space system with moving average noise', J. Franklin Inst., 2018, **355**, (6), pp. 3079-3103.

37 Zhang, X., Ding, F., Xu, L., *et al.*: 'State filtering-based least squares parameter estimation for bilinear systems using the hierarchical identification principle', IET Control Theory Appl., 2018, **12**, (12), pp. 1704-1713.

38 Ding, J.L.: 'The hierarchical iterative identification algorithm for multi-input-output-error systems with autoregressive noise'. Complexity, 2017, pp. 1-11, Article ID 5292894. doi: https://doi.org/10.1155/2017/5292894

39 Chen, M.T., Ding, F., Xu, L., *et al.*: 'Iterative identification algorithms for bilinear-in-parameter systems with autoregressive moving average noise', J. Frankl. Inst., 2017, **354**, 17, pp. 7885-7898.

40 Ding, J.L.: 'Recursive and iterative least squares parameter estimation algorithms for multiple-input-output-error systems with autoregressive noise', Circuits Syst. Signal Process., 2018, **37**, (5), pp. 1884-1906.

41 Ding, F., Wang, Y.J., Dai, J.Y., *et al.*: 'A recursive least squares parameter estimation algorithm for output nonlinear autoregressive systems using the input-output data filtering', J. Frankl. Inst., 2017, **354**, (15), pp. 6938-6955.

42 Ding, F., Meng, D.D., Dai, J.Y., *et al.*: 'Least squares based iterative parameter estimation algorithm for stochastic dynamical systems with ARMA noise using the model equivalence', Int. J. Control Autom. Syst., 2018, **16**, (2), pp. 630-639.

43 Bai, E.W.: 'A blind approach to the Hammerstein-Wiener model identification', Automatica, 2002, **38**, (6), pp. 967-979.

44 Xu, L., Ding, F., Gu, Y., *et al.*: 'A multi-innovation state and parameter estimation algorithm for a state space system with d-step state-delay', Signal Process., 2017, **140**, pp. 97-103.

45 Liu, F.: 'A note on Marcinkiewicz integrals associated to surfaces of revolution', J. Aust. Math. Soc., 2018, **104**, (3), pp. 380-402.

46 Liu, F., Xue, Q.Y., Yabuta, K.: 'Rough maximal singular integral and maximal operators supported by subvarieties on Triebel-Lizorkin spaces', Nonlinear Anal., 2018, **171**, pp. 41-72.

47 Liu, F.: 'Continuity and approximate differentiability of multisublinear fractional maximal functions', Math. Inequal. Appl., 2018, **21**, (1), pp. 25-40.

48 Yin, C.C., Wang, C.W.: 'The perturbed compound Poisson risk process with investment and debit interest', Methodol. Comput. Appl. Probab., 2010, **12**, (3), pp. 391-413.

49 Yin, C.C., Yuen, K.C.: 'Optimality of the threshold dividend strategy for the compound Poisson model', Stat. Probab. Lett., 2011, **81**, (12), pp. 1841-1846.

50 Yin, C.C., Wen, Y.Z.: 'Exit problems for jump processes with applications to dividend problems', J. Comput. Appl. Math., 2013, **245**, pp. 30-52.

51 Yin, C.C., Wen, Y.Z.: 'Optimal dividend problem with a terminal value for spectrally positive Levy processes', Insur. Math. Econ., 2013, **53**, (3), pp. 769-773.

52 Gong, P.C., Wang, W.Q., Li, F.C., *et al.*: 'Sparsity-aware transmit beamspace design for FDA-MIMO radar', Signal Process., 2018, **144**, pp. 99-103.

53 Rao, Z.H., Zeng, C.Y., Wu, M.H., *et al.*: 'Research on a handwritten character recognition algorithm based on an extended nonlinear kernel residual network', KSII Trans. Internet Inf. Syst., 2018, **12**, (1), pp. 413-435.

54 Zhao, N., Liu, R., Chen, Y., *et al.*: 'Contract design for relay incentive mechanism under dual asymmetric information in cooperative networks', Wirel. Netw., 2018. doi: 10.1007/s11276-017-1518-x

55 Zhu, D.Q., Cao, X., Sun, B., Luo, C.M.: 'Biologically inspired self-organizing map applied to task assignment and path planning of an AUV system', IEEE Trans. Cognit. Developmental Syst., 2018, **10**, (2), pp. 304-313.