Identification of Time-Varying Autoregressive Systems Using Maximum *a Posteriori* Estimation

Tesheng Hsiao, *Member, IEEE*

***Abstract*—Time-varying systems and nonstationary signals arise naturally in many engineering applications, such as speech, biomedical, and seismic signal processing. Thus, identification of the time-varying parameters is of crucial importance in the analysis and synthesis of these systems. The present time-varying system identification techniques require either demanding computation power to draw a large amount of samples (Monte Carlo-based methods) or a wise selection of basis functions (basis expansion methods). In this paper, the identification of time-varying autoregressive systems is investigated. It is formulated as a Bayesian inference problem with constraints on the conditional and prior probabilities of the time-varying parameters. These constraints can be set without further knowledge about the physical system. In addition, only a few hyper parameters need tuning for better performance. Based on these probabilistic constraints, an iterative algorithm is proposed to evaluate the maximum *a posteriori* estimates of the parameters. The proposed method is computationally efficient since random sampling is no longer required. Simulation results show that it is able to estimate the time-varying parameters reasonably well and a balance between the bias and variance of the estimation is achieved by adjusting the hyperparameters. Moreover, simulation results indicate that the proposed method outperforms the particle filter in terms of estimation errors and computational efficiency.**

***Index Terms*—Maximum *a posteriori* estimation, time-varying autoregressive model, time-varying system identification.**

I. Introduction

PARAMETRIC representations of time-varying systems and nonstationary signals are encountered frequently in various engineering applications. For example, the transitions between phonemes in speech can be modeled as time-varying autoregressive-moving-average (TV-ARMA) systems [1]; the joint effects of multipath fading channels and Doppler shifts in spread-spectrum communications are characterized by time-varying autoregressive (TVAR) systems [2]; in the study of seismic structural damage, the earthquake time histories are generated by TVAR models [3]; the event-related synchronization/desynchronization (ERS/ERD) of alpha waves of electroencephalogram (EEG) is nonstationary and is represented by a TVAR model [4]. Thus, the need for identifying the time-varying system parameters arises naturally in these areas.

Manuscript received May 31, 2007; revised January 8, 2008. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. James Lam.

The author is with the Department of Electrical and Control Engineering, National Chiao Tung University, Hsinchu 300, Taiwan, R.O.C. (e-mail: tshsiao@cn.nctu.edu.tw).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSP.2008.919393

System identification is the process of estimating parameters from the input and output data. However, most system identification techniques are developed for linear time-invariant (LTI) cases [5], [6] whereas there are relatively few studies on time-varying system identification.

The present time-varying system identification techniques can be classified into three categories. The first class of techniques is a natural extension of the well-known recursive algorithms, such as the recursive least square (RLS) algorithm. It takes advantage of the self-tuning properties of adaptive filters to estimate time-varying parameters. Since adaptive filters are designed for the purpose of online parameter estimation [7], the instantaneous value of each parameter can be identified based on the most recent input–output data, while the "old" data are discarded by incorporating the "forgetting factor" into the adaptive algorithms. Nishiyama proposed an H^{∞} optimization method to find out the "best" forgetting factor [8]. Carlos and Bershad analyzed the statistical behavior of the finite precision least mean square (LMS) adaptive filter for identification of a time-varying system [9]. However, adaptive filters can only trace "slowly-varying" parameters. For rapidly changing parameters, the performance is unsatisfactory.

In general, the identification of LTI systems' parameters is formulated as an overdetermined problem. Then, the least-square solution is the optimal estimate of the parameters in the sense of minimum residual energy. However, if the parameters vary with time, the problem becomes underdetermined and it is much more difficult to find out the "best" solution. The second class of time-varying system identification techniques resolves the underdetermined problem by expanding the time-varying parameters as a linear combination of a set of basis functions. Consequently, the unknown variables to the identification problem are transformed from a larger set of time-varying parameters into a smaller set of constant coefficients of the basis functions. Hence, the problem is solvable.

The basis functions have significant effects on the smoothness and variation speeds of the estimated parameters; however, the selection of the basis functions is application dependent and is not trivial. Commonly used basis functions include Legendre polynomials which form an orthogonal set [1], prolate spheroidal sequences that are the best approximation to bandlimited functions [10], wavelet basis, which has a distinctive property of multiresolution in both time and frequency domains [11], [12], and discrete cosine transform (DCT) that is close to the optimal Karhunen–Loeve transform [13]. Besides, regulation conditions can be easily imposed on the Fourier basis by suppressing the high-order harmonics [14]; the algebraic properties of the "complete shifted polynomials" (such as Chebyshev, Legendre, and

Laguerre polynomials) were investigated in [15] and an isomorphic matrix algebra-based method was proposed.

Additional constraints on the basis functions have been considered by other researchers. For example, Tsatsanis and Giannakis require linear independence of the "instantaneous correlation" of the basis functions [16]. They proposed a two-step method that estimates the correlations of the constant coefficients first and then extracts the coefficients by the subspace identification method. On the other hand, Kaipio and Karjalainen proposed a principal-component-analysis (PCA)-type approximation scheme to select the "optimal basis" [4]. The mutual correlations of the coefficients are also taken into account in their approach.

The basis expansion methods have been widely applied to solve various engineering problems. Eom [13] expanded a TVAR model over a set of DCT bases to analyze and classify acoustic signatures of moving vehicles. The extrema points of contours of planar shapes can be detected more accurately by expanding the TVAR parameters of the contours over a set of discrete Fourier transform (DFT) bases [14]. The accuracy of classifying high-range resolution (HRR) radar signatures is enhanced by means of TVAR models [17].

The last class of time-varying system identification techniques is based on Monte Carlo methods [18], [19] or particle filters [20]. The time-varying parameters are regarded as random variables and a large number of samples of each parameter are drawn with respect to the corresponding probability distributions. Then, the sample means are calculated as approximations to the parameters. According to the law of large numbers, the sample mean approximation approaches the true parameter provided that the number of samples is sufficiently large. Moreover, particle filters make it possible to implement the Monte Carlo approximation online. The Monte Carlo methods have the potential to solve the underdetermined problem without worries about the selection of basis functions; however, the computation power is very demanding and the computation time increases dramatically as the length of data increases.

There are still other time-varying system identification methods that do not belong to any of the aforementioned classes. For example, Bravo *et al.* proposed a set membership method to estimate time-varying parameters with guaranteed error bounds [21]. Gemez and Maravall explored the state space representation of the autoregressive-integrated-moving-average (ARIMA) models with missing observations. Then, Kalman filters were applied to estimate, predict, and interpolate the nonstationary data [22].

It can be seen from the previous discussion that each time-varying system identification method has its own strength and weakness. The advantages of adaptive filters are their solid and well-developed theoretical foundations. Thus, the performance and limits are predictable. However, the applications are restricted to slowly varying systems. On the other hand, the basis expansion approaches are able to trace rapidly changing parameters provided that appropriate basis functions are used. Unfortunately, there is no systematic way to achieve this goal. The Monte Carlo methods can also identify a wide range of classes of time-varying parameters by adjusting the underlying probabilistic assumptions. However, the algorithms tend to be computationally inefficient, especially when the "acceptance rate" of the Monte Carlo sampler is low [23].

A novel TVAR system identification method is proposed in this paper. The identification problem is formulated in a Bayesian inference framework which evaluates the posterior probability of the parameters, conditioning on the output data. Unlike Monte Carlo-based approaches, the proposed method searches the maximum of the posterior probability successively along each coordinate axis of the parameter space in an efficient way. Therefore, the time-consuming random sampling process of the Monte Carlo-based methods is avoided. Simulation studies at the end of this paper demonstrate the advantages of the proposed method over the particle filter in terms of computational efficiency and estimation accuracy. Compared with the basis expansion approaches, the proposed method can be easily and quickly implemented since additional constraints in the Bayesian inference framework are imposed "naturally" to facilitate the computation. Moreover, only a few hyperparameters need tuning in order to achieve a balance between the bias and variance of the estimation.

This paper is organized as follows. The formulation of the TVAR system identification problem as well as the probabilistic assumptions is presented in Section II. The iterative procedure for maximizing the posterior probability is proposed in Section III. Simulations are conducted in Section IV. Section V concludes this paper.

II. Problem Formulation

A. TVAR Systems and Assumptions

An nth order time-varying autoregressive system can be expressed as follows:

$$y(k) = -\sum_{i=1}^{n} a_i(k) y(k-i) + \varepsilon(k) \tag{1}$$

where $y(k) \in \mathbb{R}$ is the output sequence and $\varepsilon(k)$ is the process noise. $\varepsilon(k)$ is assumed to be a zero mean Gaussian distributed white noise with variance σ_ε^2 for all k. $a_i(k)$'s are the system parameters to be estimated.

Suppose that the order n is known and a set of $(N+n)$-point output data $\{y(-n), y(-n+1), \ldots, y(0), y(1), \ldots, y(N-1)\}$ has been collected. System identification concerns the problem of estimating the $n \cdot N$ parameters $a_i(k)$, $i = 1, \ldots, n$, and $k = 0, \ldots, N-1$ from the set of output data. Unfortunately, solving (1) for all $a_i(k)$'s directly is difficult, if not impossible, since it is an underdetermined problem; moreover, the process noise $\varepsilon(k)$ and its variance σ_ε^2 are unknown.

There are two ways to tackle the underdetermined problem. One is to reduce the number of unknown variables while the other is to impose more constraints on the parameters. Basis expansion approaches belong to the former. If the parameters $a_i(k)$'s are expanded over a set of basis functions, the unknown variables of the identification problem are reduced to a smaller set of constant coefficients of the basis functions. However, different basis functions result in different estimates of the param-

eters and the selection of the "optimal" basis functions is not trivial.

On the other hand, additional constraints can be explicitly imposed on the parameters $a_i(k)$'s. To set up these additional constraints, each $a_i(k)$ is treated as a random variable. Then, Bayesian inference provides a framework for imposing constraints on random variables in the form of conditional distributions and prior distributions [24], [25]. The posterior probability derived from Bayes's theorem [24] is a candidate of the optimum criterion for the choice of the best estimates of $a_i(k)$'s. In order not to cause confusion about the notations in use, from now on, let $a_i(k)$ denote a random variable while $a_i^*(k)$ is the instance of $a_i(k)$ that satisfies (1) (i.e., $a_i^*(k)$ holds the true value of the TVAR system's parameter), given the set of output data $\{y(k)|k=-n,1,\ldots,N-1\}$.

Since all $a_i(k)$'s are random variables, their probability distributions, either joint probabilities or conditional probabilities, need to be specified as parts of the Bayesian inference framework. If the differences of $a_i(k)$'s at two consecutive time steps do not vary significantly (i.e., $R_i = \max_{0\le k<N-1} |a_i(k+1) - a_i(k)| / \min_{0\le k<N-1} |a_i(k+1) - a_i(k)|$ is not too large), it is reasonable to assume that $a_i(k+1)$ stays in an "equal-sized" neighborhood of $a_i(k)$ for all k. Therefore, the following assumptions are made:

$$a_i(k+1) \sim N\left(\cdot|a_i(k),\sigma_i^2\right), \quad i=1,\ldots,n; k=0,\ldots,N-2 \tag{2}$$

$$a_i(0) \sim N\left(\cdot|\mu_i,\sigma_i^2\right), \quad i=1,2,\ldots,n \tag{3}$$

$$a_i(k)||a_j(l), \quad i,j=1,\ldots,n, i\neq j; k,l=0,\ldots,N-1 \tag{4}$$

where $N(\cdot|\mu,\sigma^2)$ denotes the Gaussian distribution with mean μ and variance σ^2. $a_i(k)||a_j(l)$ denotes that $a_i(k)$ and $a_j(l)$ are independent. μ_i's and σ_i's in (2) and (3) are parameters of Gaussian distributions. σ_i's are also regarded as random variables and are endowed with their own probability distributions. Detailed discussions will be given in the remainder of this section.

Remarks:

1) Equations (2) and (3) assume that $a_i(k+1)$ is around $a_i(k)$. The smaller σ_i is, the more likely that $a_i(k+1)$ is close to $a_i(k)$. These assumptions are not very restrictive because no direction preference of $a_i(k+1)$ is implied by (2) and (3) (i.e., $a_i(k+1)$ may be either larger or smaller than $a_i(k)$ with equal probability). Hence, the fast variation of $a_i(k)$ is allowed under these assumptions. However, these assumptions do require that all $a_i(k)$'s, $k=0,\ldots,N-1$ vary in a "uniform" way (i.e., $R_i \approx 1$), because the variances in (2) and (3) are the same for all k.
2) It is arguable that the independency assumption of (4) is valid since intercorrelations among parameters may be significant in some physical systems. However, the assumption of (4) considerably simplifies the Bayesian inference and results in an elegant algorithm. In addition, simulations in Section IV show that even though the parameters are intercorrelated, the algorithm based on the independency assumption of (4) still yields a satisfactory result.

Although additional constraints are imposed on $a_i(k)$'s by (2)–(4), new parameters μ_i's and σ_i's are introduced. Since there is no clue about the values or ranges of σ_i's, they are assumed to be random variables and, thus, a hierarchical structure of random variables is established [24]. The probability distributions of σ_i's, called the prior distributions, reflect the designer's subjective belief in σ_i's and are somewhat arbitrary. For example, the Jeffrey's prior is said to be noninformative, representing a lack of prior knowledge about σ_i's. On the other hand, conjugate priors simplify the mathematical expression of the posterior probability density function [24].

It will be clearer later that the subsequent derivations become much easier if the prior distributions are assigned to σ_i^{-2}'s, instead of σ_i's. It is assumed that σ_i^{-2} has a conjugate prior distribution of Gaussian distribution (i.e., the Gamma distribution)

$$\begin{aligned}\frac{1}{\sigma_i^2} &\sim Ga\left(\sigma_i^{-2}|\alpha_i,\beta_i^{-1}\right)\\ &\triangleq \frac{1}{\Gamma(\alpha_i)\beta_i^{-\alpha_i}}\left(\frac{1}{\sigma_i^2}\right)^{\alpha_i-1}\exp\left(\frac{-\beta_i}{\sigma_i^2}\right)\end{aligned} \tag{5}$$

where $\alpha_i,\ \beta_i > 0$, $i=1,\ldots,n$, and $\Gamma(\alpha_i)$ are the Gamma function.

Again, new parameters α_i's and β_i's are introduced in (5). They are called hyperparameters because they are parameters of the prior distributions. We can construct a new layer of hierarchy for α_i's and β_i's by treating them as random variables and assign probability distributions to them. This will, in turn, introduce more hyperparameters in the distributions of α_i's and β_i's. The same procedure can be repeated as many times as we want since there is no limit on the number of hierarchies that can be built in the Bayesian framework; however, there is no benefit in using a complicated hierarchic structure. Therefore, α_i's and β_i's will not be modeled as random variables. Instead, specific values will be assigned to them.

Similarly, σ_ε is treated as a random variable and σ_ε^{-2} is assumed to be Gamma distributed, i.e.,

$$\frac{1}{\sigma_\varepsilon^2} \sim Ga\left(\cdot|\alpha_\varepsilon,\beta_\varepsilon^{-1}\right) \tag{6}$$

where the hyperparameters α_ε and β_ε will be assigned specific values.

Equations (5) and (6) assign prior distributions to σ_i^{-2} and σ_ε^{-2}, respectively; however, the relationship among these random variables in terms of joint probabilities or conditional probabilities remains unspecified. It is reasonable to assume that σ_i^{-2} and σ_ε^{-2} are mutually independent. Thus, we have

$$\sigma_i^{-2}||\sigma_j^{-2}, \quad i,j=1,\ldots,n, i\neq j \tag{7}$$

$$\sigma_i^{-2}||\sigma_\varepsilon^{-2}, \quad i=1,\ldots,n. \tag{8}$$

The parameters μ_i's introduced in (3) are the means of $a_i(0)$'s. Since we have assumed that the variances of $a_i(k)$ conditioning on $a_i(k-1)$ are the same for all k, the values of μ_i's can be inferred from $a_i(1)$ and/or $a_i(2)$. In order not to introduce more parameters into the probabilistic assumptions, μ_i's are assumed to be deterministic parameters whose values will be determined later.

Remark: Equations (2)–(8) constitute the additional constraints that are imposed on the TVAR system's parameters. Then the posterior probability of $a_i(k)$ conditioning on $y(k)$ will be derived from these constraints and serve as an optimum criterion for the evaluation of the best estimate of $a_i(k)$. Note that it is possible to make other sets of assumptions other than (2)–(8). For example, Godsill and Clapp assumed that the process noise $\varepsilon(k)$ is nonstationary and the conditional distribution of its variance satisfies the following [20]:

$$p\left(\phi_\varepsilon(k)|\phi_\varepsilon(k-1),\sigma^2_{\phi_\varepsilon}\right)=N\left(\phi_\varepsilon(k)|\phi_\varepsilon(k-1),\sigma^2_{\phi_\varepsilon}\right) \quad (9)$$

where $\phi_\varepsilon(k)=\log(\sigma_\varepsilon(k))$. In addition, the conditional distributions of parameters $a_i(k)$ are assumed to be

$$p\left(a_i(k)|a_i(k-1),\sigma^2_a\right)=N\left(a_i(k)|\alpha a_i(k-1),\sigma^2_a\right)$$

where $0<\alpha<1$. The hyperparameters σ_a, $\sigma_{\phi_\varepsilon}$, and α are constants with prespecified values. In comparison with (6), (9) takes into account the nonstationary property of the process noise. However, more random variables (i.e., $\phi_\varepsilon(k)$'s), are introduced, resulting in a more complicated algorithm and more demanding computation.

B. Posterior Probability

First, the following vector notations are defined for easy reference later:

$$\begin{aligned}
\mathbf{y}&=[y(0),y(1),\dots y(N-1)]^{\mathrm{T}}\in\mathbb{R}^N\\
\mathbf{a}_i&=[a_i(0),a_i(1),\dots,a_i(N-1)]^{\mathrm{T}}\in\mathbb{R}^N,\quad i=1,\dots,n\\
\mathbf{a}&=\left[\mathbf{a}_1^{\mathrm{T}},\dots,\mathbf{a}_n^{\mathrm{T}}\right]^{\mathrm{T}}\in\mathbb{R}^{N\cdot n}\\
\boldsymbol{\varepsilon}&=[\varepsilon(0),\varepsilon(1),\dots,\varepsilon(N-1)]^{\mathrm{T}}\in\mathbb{R}^N\\
\boldsymbol{\sigma}^{-2}&=\left[\sigma_1^{-2},\quad\sigma_2^{-2},\quad\cdots\quad\sigma_n^{-2}\right]^T\in\mathbb{R}^n\\
\mathbf{Y}_i&=\mathrm{diag}\left(y(-i),\dots,y(-1),y(0),\dots,y(N-i-1)\right)\\
&\quad\in\mathbb{R}^{N\times N},\quad i=1,\dots,n
\end{aligned}$$

where $\mathrm{diag}(\cdot)$ denotes a diagonal matrix whose diagonal elements are listed inside the parentheses. In addition, $\mathbf{a}_i^*$ and $\hat{\mathbf{a}}_i$ denote the true and estimated values of $\mathbf{a}_i$, $i=1,\dots,\mathrm{n}$, respectively. Similarly, $\mathbf{a}^*$ and $\hat{\mathbf{a}}$ denote the true and estimated values of $\mathbf{a}$, respectively.

In this subsection, we are going to derive the posterior probability density function $p(\mathbf{a}|\mathbf{y})$ based on the TVAR system (1) and assumptions of (2)–(8). Then, the optimal estimate of $\mathbf{a}$ will be the one that maximizes the posterior probability $p(\mathbf{a}|\mathbf{y})$.

The posterior probability density function $p(\mathbf{a}|\mathbf{y})$ can be marginalized from $p(\mathbf{a},\sigma_\varepsilon^{-2},\boldsymbol{\sigma}^{-2}|\mathbf{y})$, which by Bayes's theorem is

$$\begin{aligned}
p\left(\mathbf{a},\sigma_\varepsilon^{-2},\boldsymbol{\sigma}^{-2}|\mathbf{y}\right)&\propto p\left(\mathbf{y}|\mathbf{a},\sigma_\varepsilon^{-2},\boldsymbol{\sigma}^{-2}\right)\\
&\quad\times p\left(\mathbf{a}|\sigma_\varepsilon^{-2},\boldsymbol{\sigma}^{-2}\right)p\left(\sigma_\varepsilon^{-2},\boldsymbol{\sigma}^{-2}\right). \quad (10)
\end{aligned}$$

According to the assumption of (8) and the fact that $\varepsilon(k)$ is white noise, the first term on the right-hand side of (10) is

$$\begin{aligned}
&p\left(\mathbf{y}|\mathbf{a},\sigma_\varepsilon^{-2},\boldsymbol{\sigma}^{-2}\right)\\
&\quad=\prod_{k=0}^{N-1}p\left(y(k)|a_1(k),\dots,a_n(k),\sigma_\varepsilon^{-2}\right)\\
&\quad\propto\left(\sigma_\varepsilon^{-2}\right)^{\frac{N}{2}}\exp\left(\frac{-1}{2\sigma_\varepsilon^2}\sum_{k=0}^{N-1}\left(y(k)+\sum_{i=1}^{n}a_i(k)y(k-i)\right)^2\right).
\end{aligned}$$

Equations (2)–(4), (7), and (8) give the expression of the second term on the right side of (10)

$$\begin{aligned}
&p\left(\mathbf{a}|\sigma_\varepsilon^{-2},\boldsymbol{\sigma}^{-2}\right)\\
&\quad=\prod_{i=1}^{n}\left[\prod_{k=1}^{N-1}p\left(a_i(k)|a_i(k-1),\sigma_i^{-2}\right)\right]p\left(a_i(0)|\sigma_i^{-2}\right)\\
&\quad\propto\prod_{i=1}^{n}\left(\sigma_i^{-2}\right)^{\frac{N}{2}}\exp\left(\frac{-1}{2\sigma_i^2}\left[\sum_{k=1}^{N-1}\left(a_i(k)-a_i(k-1)\right)^2\right.\right.\\
&\qquad\left.\left.+\left(a_i(0)-\mu_i\right)^2\right]\right).
\end{aligned}$$

Assumptions about the prior distributions [(6) and (8)] yield the following expression of the third term on the right-hand side of (10):

$$\begin{aligned}
p\left(\sigma_\varepsilon^{-2},\boldsymbol{\sigma}^{-2}\right)&=p\left(\sigma_\varepsilon^{-2}\right)\prod_{i=1}^{n}p\left(\sigma_i^{-2}\right)\\
&\propto\left(\sigma_\varepsilon^{-2}\right)^{\alpha_\varepsilon-1}\exp\left(\frac{-\beta_\varepsilon}{\sigma_\varepsilon^2}\right)\\
&\quad\cdot\prod_{i=1}^{n}\left(\sigma_i^{-2}\right)^{\alpha_i-1}\exp\left(\frac{-\beta_i}{\sigma_i^2}\right).
\end{aligned}$$

Therefore, (10) becomes

$$\begin{aligned}
&p\left(\mathbf{a},\sigma_\varepsilon^{-2},\boldsymbol{\sigma}^{-2}|\mathbf{y}\right)\\
&\quad\propto\left(\sigma_\varepsilon^{-2}\right)^{\frac{N}{2}+\alpha_\varepsilon-1}\\
&\qquad\times\exp\left(\frac{-1}{2\sigma_\varepsilon^2}\left[\sum_{k=0}^{N-1}\left(y(k)+\sum_{i=1}^{n}a_i(k)y(k-i)\right)^2+2\beta_\varepsilon\right]\right)\\
&\qquad\times\prod_{i=1}^{n}\left(\sigma_i^{-2}\right)^{\frac{N}{2}+\alpha_i-1}\\
&\qquad\times\exp\left(\frac{-1}{2\sigma_i^2}\left[\sum_{k=1}^{N-1}\left(a_i(k)-a_i(k-1)\right)^2\right.\right.\\
&\qquad\left.\left.+\left(a_i(0)-\mu_i\right)^2+2\beta_i\right]\right).
\end{aligned}$$

Marginalizing this density function (i.e., integrating out σ_ε^{-2} and σ_i^{-2}'s), gives the desired posterior probability density function $p(\mathbf{a}|\mathbf{y})$. Namely

$$\begin{aligned} p(\mathbf{a}|\mathbf{y}) &= \int_0^\infty \int_0^\infty p\left(\mathbf{a}, \sigma_\varepsilon^{-2}, \boldsymbol{\sigma}^{-2}|\mathbf{y}\right) d\left(\sigma_\varepsilon^{-2}\right) d(\boldsymbol{\sigma}^{-2}) \\ &\propto \left\{ \int_0^\infty p\left(\mathbf{y}|\mathbf{a}, \sigma_\varepsilon^{-2}\right) p\left(\sigma_\varepsilon^{-2}\right) d\left(\sigma_\varepsilon^{-2}\right) \right\} \\ &\quad \times \left\{ \prod_{i=1}^n \int_0^\infty p\left(\mathbf{a}_i|\sigma_i^{-2}\right) p\left(\sigma_i^{-2}\right) d\left(\sigma_i^{-2}\right) \right\}. \end{aligned}$$

The integrations are carried out by repeatedly applying integration by parts. By straightforward calculation, the posterior probability density function is

$$p(\mathbf{a}|\mathbf{y}) \propto g(\mathbf{a}|\mathbf{y}) \prod_{i=1}^n f_i(\mathbf{a}_i) \tag{11}$$

where

$$g(\mathbf{a}|\mathbf{y}) = \left[\frac{1}{2} \sum_{k=0}^{N-1} \left(y(k) + \sum_{i=1}^n a_i(k) y(k-i) \right)^2 + \beta_\varepsilon \right]^{-p_\varepsilon} \tag{12}$$

and

$$\begin{aligned} f_i(\mathbf{a}_i) = \Bigg[&\frac{1}{2} \sum_{k=1}^{N-1} \left(a_i(k) - a_i(k-1) \right)^2 \\ &+ \frac{1}{2} \left(a_i(0) - \mu_i \right)^2 + \beta_i \Bigg]^{-p_i}. \end{aligned} \tag{13}$$

$p_\varepsilon = (N/2) + \alpha_\varepsilon$ and $p_i = (N/2) + \alpha_i$. The hyperparameters α_ε and α_i's are chosen such that $p_\varepsilon = p_i = p \in \mathbb{N}$, for $i = 1, \ldots, n$ in order to facilitate the maximization process later [see the remark after (18)].

Various optimum criteria can be established based on the posterior probability $p(\mathbf{a}|\mathbf{y})$. For example, the conditional expected value $\mathrm{E}[\mathbf{a}|\mathbf{y}]$ is known to be the minimum variance estimate of $\mathbf{a}$. On the other hand, the maximizer of $p(\mathbf{a}|\mathbf{y})$ is another optimal estimate of $\mathbf{a}$ in the sense that it holds the most likely value of $\mathbf{a}$ conditioning on the output data. The maximum *a posteriori* estimate will be investigated in the next section.

Remark: Stability of the estimated system is not guaranteed under current problem setting and assumptions. Actually, the notion of stability of time-varying systems is very subtle. It cannot be easily checked from its parameters or instantaneous pole locations. The condition that all poles reside inside the unit circle at each time step does not guarantee stability of the linear time-varying system. Detailed discussions about the stability of time-varying systems can be found in [26].

III. Maximum *a Posteriori* Estimation

A. Iterative Update of the MAP Estimate

It is well known that the conditional expected value $\mathrm{E}[\mathbf{a}|\mathbf{y}]$ is the minimum variance estimate of $\mathbf{a}$ given $\mathbf{y}$. However, the closed form of this conditional mean is not available in view of the complicated structure of the posterior probability $p(\mathbf{a}|\mathbf{y})$ (11)–(13). Many researchers have applied the Monte Carlo method or its variants to evaluate the conditional mean numerically. However, drawing a large amount of samples of $\mathbf{a}$, which is a necessary step for all Monte Carlo-based methods, from complicated density functions such as (11)–(13) is time consuming. Its efficiency deteriorates as the length of data increases. To get rid of the inefficient sampling process, this paper evaluates the maximum *a posteriori* (MAP) estimate of $\mathbf{a}$.

The MAP estimate, denoted by $\hat{\mathbf{a}}_{\mathrm{MAP}}$, is the maximizer of the posterior probability density function $p(\mathbf{a}|\mathbf{y})$ (i.e. $\hat{\mathbf{a}}_{\mathrm{MAP}} = \arg\max_{\mathbf{a}} p(\mathbf{a}|\mathbf{y})$). The reason to choose $\hat{\mathbf{a}}_{\mathrm{MAP}}$ as an estimate of $\mathbf{a}$ is that it is the most likely value of $\mathbf{a}$ given the observed output $\mathbf{y}$. Since $\hat{\mathbf{a}}_{\mathrm{MAP}}$ maximizes $p(\mathbf{a}|\mathbf{y})$, the first derivative of $p(\mathbf{a}|\mathbf{y})$ must vanish at $\hat{\mathbf{a}}_{\mathrm{MAP}}$, i.e.,

$$\left. \frac{\partial p(\mathbf{a}|\mathbf{y})}{\partial a_i(k)} \right|_{\mathbf{a}=\hat{\mathbf{a}}_{\mathrm{MAP}}} = 0, \quad i = 1, \ldots, n, \text{ and } k = 0, \ldots, N-1. \tag{14}$$

Equation (14) consists of $n \cdot N$ nonlinear equations and it is intractable to solve these equations simultaneously to obtain $\hat{\mathbf{a}}_{\mathrm{MAP}}$. Instead, an iterative procedure is proposed that manipulates one variable at a time and lets all of the others hold their values from the previous iteration. Suppose that $\hat{\mathbf{a}}^{(j)} \in \mathbb{R}^{n \cdot N}$ is obtained as an approximation of $\hat{\mathbf{a}}_{\mathrm{MAP}}$ at the jth iteration. Then, the elements of $\hat{\mathbf{a}}^{(j)}$ are updated one by one into $\hat{\mathbf{a}}^{(j+1)}$. For each update, only one variable in one equation of (14) needs to be taken care of; hence, the complexity of the problem is reduced. The procedure goes on iteratively in a way that drives $\hat{\mathbf{a}}^{(j)}$ to the local maximum of $p(\mathbf{a}|\mathbf{y})$ as j approaches infinity.

For easy reference, let us define the permutation function $\tau : \mathbb{R}^{N \cdot n} \to \mathbb{R}^{N \cdot n}$ which permutes $\mathbf{a}$ in the order that $\hat{\mathbf{a}}^{(j)}$ is updated, namely

$$\begin{aligned} \tau(\mathbf{a}) = [&a_1(0), \ldots, a_n(0), a_1(1), \ldots, a_n(1), \ldots, \\ &a_1(N-1), \ldots, a_n(N-1)]^{\mathrm{T}}. \end{aligned}$$

Let $\tau_s(\mathbf{a}) \in \mathbb{R}$ denote the sth element of $\tau(\mathbf{a})$. It is clear from the definition of $\tau(\mathbf{a})$ that $a_i(k) = \tau_{kn+i}(\mathbf{a})$. Besides, let $\tau_{s:t}(\mathbf{a}) = [\tau_s(\mathbf{a}), \tau_{s+1}(\mathbf{a}), \ldots, \tau_t(\mathbf{a})]^{\mathrm{T}}$ be a subvector of $\tau(\mathbf{a})$ for $1 \le s \le t \le N \cdot n$.

Suppose that at the jth iteration, $a_i(k)$ is going to be updated for some i, k. $p(\mathbf{a}|\mathbf{y})$ in (11)–(13) is rewritten as a function of that single variable $a_i(k)$. Hence (13) becomes

$$\begin{aligned} f_i(\mathbf{a}_i) &= f_{i,k}\left(a_i(k)|a_i(0), \ldots, a_i(k-1), \right. \\ &\qquad \left. a_i(k+1), \ldots, a_i(N-1)\right) \\ &\triangleq \left\{ \delta(k) \eta_i^2(k) \left[1 + \left(\frac{a_i(k) - c_i(k)}{\eta_i(k)} \right)^2 \right] \right\}^{-p} \end{aligned} \tag{15}$$

where $\delta(N-1)=0.5$ and $\delta(k)=1$ for $k=0,1,\ldots,N-2$

$$c_i(k)=\begin{cases} a_i(N-2), & k=N-1 \\ \frac{1}{2}\left(a_i(k-1)+a_i(k+1)\right), & 0<k<N-1 \\ \frac{1}{2}\left(a_i(1)+\mu_i\right), & k=0 \end{cases}$$

(see the equation shown at the bottom of the page).

Equation (12) becomes

$$g(\mathbf{a}|\mathbf{y})=g_{i,k}\left(a_i(k)|\tau_{1:s-1}(\mathbf{a}),\tau_{s+1:N\cdot n}(\mathbf{a})\right),\quad s=kn+i \triangleq \begin{cases} \left\{\frac{y^2(k-i)}{2}\overline{\eta}_i^2(k) \times\left[1+\left(\frac{a_i(k)-\overline{c}_i(k)}{\overline{\eta}_i(k)}\right)^2\right]\right\}^{-p}, & y(k-i)\neq 0 \\ 1, & y(k-i)=0 \end{cases} \tag{16}$$

where

$$\overline{c}_i(k)=\frac{-1}{y(k-i)}\left(y(k)+\sum_{l=1,l\neq i}^{n}a_l(k)y(k-l)\right)$$

and

$$\overline{\eta}_i(k)=\frac{1}{|y(k-i)|}\times\left[\sum_{m=0,m\neq k}^{N-1}\left(y(m)+\sum_{l=1}^{n}a_l(m)y(m-l)\right)^2+2\beta_\varepsilon\right]^{\frac{1}{2}}.$$

Note that if $y(k-i)=0$, $\overline{c}_i(k)$ and $\overline{\eta}_i(k)$ are not well defined. But in this case, $g(\mathbf{a}|\mathbf{y})$ is independent of $a_i(k)$ (12) and becomes a constant with respect to $a_i(k)$. In other words

$$p(\mathbf{a}|\mathbf{y})\propto f_{i,k}\left(a_i(k)|a_i(0),\ldots,a_i(k-1),a_i(k+1),\ldots,a_i(N-1)\right)$$

Thus, define $g_{i,k}(a_i(k)|\tau_{1:s-1}(\mathbf{a}),\tau_{s+1:N\cdot n}(\mathbf{a}))=1$ whenever $y(k-i)=0$.

Substituting (15) and (16) into (14), we have

$$\frac{d}{d\left(a_i(k)\right)}\left\{\left[1+\left(\frac{a_i(k)-c_i(k)}{\eta_i(k)}\right)^2\right]\times\left[1+\left(\frac{a_i(k)-\overline{c}_i(k)}{\overline{\eta}_i(k)}\right)^2\right]\right\}^{-p}=0 \tag{17}$$

for $i=1,\ldots,n$ and $k=0,\ldots,N-1$.

By straightforward calculation, (17) is equivalent to

$$a^3-\frac{3}{2}(c+\overline{c})a^2+\frac{1}{2}(c^2+\overline{c}^2+4c\overline{c}+\eta^2+\overline{\eta}^2)a-\frac{1}{2}\left[c(\overline{c}^2+\overline{\eta}^2)+\overline{c}(c^2+\eta^2)\right]=0. \tag{18}$$

In (18), the subscript i and the time index k of $a_i(k)$, $c_i(k)$, $\eta_i(k)$, $\overline{c}_i(k)$, and $\overline{\eta}_i(k)$ are dropped in order for a simple expression. Note that the left-hand side of (18) is a third-order polynomial. At least one of its three roots must be real and these roots can be found analytically without applying numerical methods; thus, the calculation of the roots can be accomplished efficiently.

Remark: Now it is clear to see why we choose α_ε and α_i such that $p=(N/2)+\alpha_\varepsilon=(N/2)+\alpha_i\in\mathbb{N}$. By doing so, the maximization process is reduced to the problem of solving the roots of a third-order polynomial. Otherwise, time-consuming numerical methods would be required to find out the maximizer of $p(\mathbf{a}|\mathbf{y})$ and the resulting algorithm loses its computational efficiency.

Let r_m, $1\leq m\leq 3$ be the real solutions to (18) at the jth iteration. If

$$r=\arg\max_m p\left(\tau_{1:s-1}\left(\hat{\mathbf{a}}^{(j)}\right),r_m,\tau_{s+1:N\cdot n}\left(\hat{\mathbf{a}}^{(j-1)}\right)|\mathbf{y}\right)\quad s=kn+i \tag{19}$$

then take r as an estimate of $a_i(k)$. The maximization process of (19) just compares, at most, three values and picks up the largest one. Therefore, it can be done very quickly.

At each iteration, (18) and (19) are solved successively for $i=1,\ldots,n$ and $k=0,\ldots,N-1$. When $a_i(k)$ is updated, other random variables $a_m(l)$, $m\neq i$ and $l\neq k$ hold their most recent values. The operations performed at each iteration are summarized in Algorithm I.

$$\eta_i(k)=\begin{cases} \left[\sum_{l=1}^{N-2}\left(a_i(l)-a_i(l-1)\right)^2+\left(a_i(0)-\mu_i\right)^2+2\beta_i\right]^{\frac{1}{2}}, & k=N-1 \\ \left[\frac{1}{4}\left(a_i(k-1)-a_i(k+1)\right)^2+\frac{1}{2}\sum_{l=1,l\neq k}^{k-1}\left(a_i(l)-a_i(l-1)\right)^2+\frac{1}{2}\left(a_i(0)-\mu_i\right)^2+\beta_i\right]^{\frac{1}{2}}, & 0<k<N-1 \\ \left[\frac{1}{4}\left(a_i(1)-\mu_i\right)^2+\frac{1}{2}\sum_{l=2}^{N-1}\left(a_i(l)-a_i(l-1)\right)^2+\beta_i\right]^{\frac{1}{2}}. & k=0 \end{cases}$$

Algorithm I: MAP estimate

Given hyperparameters β_ε, β_i, and μ_i, $i = 1, \ldots, n$, let $\hat{\mathbf{a}}^{(j)}$ be the estimate of $\mathbf{a}$ obtained at the jth iteration.

At the $(j+1)$th iteration

For $k = 0, 1, \ldots, N-1$ {

For $i = 1, 2, \ldots, n$ {

Step 1) Calculate $c_i(k)$ and $\eta_i(k)$ in $f_{i,k}(a_i(k)|\hat{a}_i^{(j+1)}(0), \ldots, \hat{a}_i^{(j+1)}(k-1), \hat{a}_i^{(j)}(k+1), \ldots, \hat{a}_i^{(j)}(N-1))$.

Step 2) If $y(k-i) = 0$,

$r = c_i(k)$ and go to Step 7)

else go to Steps 3)–7).

Step 3) $s = kn + i$

Step 4) Calculate $\overline{c}_i(k)$ and $\overline{\eta}_i(k)$ in $g_{i,k}(a_i(k)|\tau_{1:s-1}(\hat{\mathbf{a}}^{(j+1)}), \tau_{s+1:Nn}(\hat{\mathbf{a}}^{(j)}))$.

Step 5) Find r_m, $1 \le m \le 3$, the real solution(s) to (18).

Step 6) Find r in (19).

Step 7) $\hat{a}_i^{(j+1)}(k) = r$ }

}.

Algorithm I is a "coordinate climbing" maximization procedure. $\hat{\mathbf{a}}^{(j)}$ moves toward the maximal point along each coordinate axis of the parameter space. For each move, (19) guarantees that $\hat{\mathbf{a}}^{(j)}$ reaches the maximal point along the axis of $a_i(k)$; thus, $p(\tau(\hat{\mathbf{a}}^{(j)})|\mathbf{y})$ is nondecreasing as j increases. This can be shown easily as follows:

$$\begin{aligned} p\left(\tau\left(\hat{\mathbf{a}}^{(j)}\right)|\mathbf{y}\right) &\le p\left(\tau_1\left(\hat{\mathbf{a}}^{(j+1)}\right), \tau_{2:Nn}\left(\hat{\mathbf{a}}^{(j)}\right)|\mathbf{y}\right) \\ &\le p\left(\tau_{1:2}\left(\hat{\mathbf{a}}^{(j+1)}\right), \tau_{3:Nn}\left(\hat{\mathbf{a}}^{(j)}\right)|\mathbf{y}\right) \\ &\le \ldots \le p\left(\tau\left(\hat{\mathbf{a}}^{(j+1)}\right)|\mathbf{y}\right). \end{aligned}$$

Since $p(\tau(\hat{\mathbf{a}}^{(j)})|\mathbf{y})$ is nondecreasing and $p(\mathbf{a}|\mathbf{y})$ is upper bounded, it follows that $p(\tau(\hat{\mathbf{a}}^{(j)})|\mathbf{y})$ converges to the local maximum of $p(\mathbf{a}|\mathbf{y})$. Note that $\hat{\mathbf{a}}_{\mathrm{MAP}}$ is the global maximum of $p(\mathbf{a}|\mathbf{y})$; however, the proposed method only guarantees reaching the local maximum.

B. Selection of the Hyperparameters

Algorithm I illustrates an iterative method to approximate the MAP estimate asymptotically, given the prespecified hyperparameters β_ε, β_i's, and μ_i's. This subsection explores the issue of setting up these hyperparameters.

μ_i is the mean of $a_i(0)$ (3). It is unknown and needs estimating. Therefore, μ_i is set as its maximum-likelihood estimate $\hat{\mu}_{i,ML}$. In other words, the first derivative of the log-likelihood function $\log p(\mathbf{a}|\mathbf{y}, \mu_i)$, with respect to μ_i, must vanish at $\hat{\mu}_{i,ML}$. Here, μ_i is written explicitly as an argument of $p(\mathbf{a}|\mathbf{y})$ to emphasize that it is the argument of the maximization.

The first derivative of the log-likelihood function is

$$\begin{aligned} 0 &= \frac{\partial}{\partial \mu_i} \log p(\mathbf{a}|\mathbf{y}, \mu_i) \\ &\propto \frac{\partial}{\partial \mu_i} f_i(\mathbf{a}_i|\mu_i) \\ &\propto \frac{-(a_i(0) - \mu_i)}{\frac{1}{2}\sum_{k=1}^{N-1}(a_i(k) - a_i(k-1))^2 + \frac{1}{2}(a_i(0) - \mu_i)^2 + \beta_i}. \end{aligned}$$

Thus, $\hat{\mu}_{i,\mathrm{ML}} = a_i(0)$. Since $a_i(0)$ is unknown, it is replaced by the estimated value $\hat{a}_i^{(j)}(0)$ at the jth iteration. Then, $\hat{\mu}_{i,ML}$ is updated along with $\hat{a}_i^{(j)}(0)$.

Now consider the values of hyperparameters β_ε and β_i's. First, let us rewrite the TVAR system's equation $g(\mathbf{a}|\mathbf{y})$ and $f_i(\mathbf{a}_i)$ [i.e., (1), (12), and (13), respectively] in terms of the matrix notations defined in Section II-B

$$\mathbf{y} = -\sum_{i=1}^{n} \mathbf{Y}_i \mathbf{a}_i^* + \boldsymbol{\varepsilon} \tag{20}$$

$$g(\mathbf{a}|y) = \left[\frac{1}{2}\left(\mathbf{y} + \sum_{i=1}^{n}\mathbf{Y}_i\mathbf{a}_i\right)^T\left(\mathbf{y} + \sum_{i=1}^{n}\mathbf{Y}_i\mathbf{a}_i\right) + \beta_\varepsilon\right]^{-p} \tag{21}$$

$$f_i(\mathbf{a}_i) = \left[\frac{1}{2}\mathbf{a}_i^T\mathbf{D}\mathbf{a}_i - a_i(0)\mu_i + \frac{1}{2}\mu_i^2 + \beta_i\right]^{-p} \tag{22}$$

where $\mathbf{D}$ is an $N \times N$ tridiagonal matrix

$$\mathbf{D} = \begin{bmatrix} 2 & -1 & 0 & 0 & \cdots & 0 \\ -1 & 2 & -1 & 0 & \cdots & 0 \\ 0 & -1 & 2 & -1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & -1 & 2 & -1 \\ 0 & 0 & \cdots & 0 & -1 & 1 \end{bmatrix}.$$

From (20)–(22), (14) is equivalent to

$$\begin{aligned} \mathbf{0} = &\left[\mathbf{Y}_i\left(\mathbf{y} + \sum_{m=1}^{n}\mathbf{Y}_m\hat{\mathbf{a}}_{m,\mathrm{MAP}}\right)\right] \\ &\times \prod_{m=1}^{n}\left[\frac{1}{2}\hat{\mathbf{a}}_{m,\mathrm{MAP}}^T\mathbf{D}\hat{\mathbf{a}}_{m,\mathrm{MAP}} - \hat{\mathbf{a}}_{m,\mathrm{MAP}}(0)\mu_m + \frac{1}{2}\mu_m^2 + \beta_m\right] + [\mathbf{D}\hat{\mathbf{a}}_{i,\mathrm{MAP}} - \mu_i\mathbf{e}_1] \\ &\times \prod_{\substack{m=1 \\ m\neq i}}^{n}\left[\frac{1}{2}\hat{\mathbf{a}}_{m,\mathrm{MAP}}^T\mathbf{D}\hat{\mathbf{a}}_{m,\mathrm{MAP}} - \hat{\mathbf{a}}_{m,\mathrm{MAP}}(0)\mu_m + \frac{1}{2}\mu_m^2 + \beta_m\right] \\ &\times \left[\frac{1}{2}\left(\mathbf{y} + \sum_{m=1}^{n}\mathbf{Y}_m\hat{\mathbf{a}}_{m,\mathrm{MAP}}\right)^T\left(\mathbf{y} + \sum_{m=1}^{n}\mathbf{Y}_m\hat{\mathbf{a}}_{m,\mathrm{MAP}}\right) + \beta_\varepsilon\right] \end{aligned} \tag{23}$$

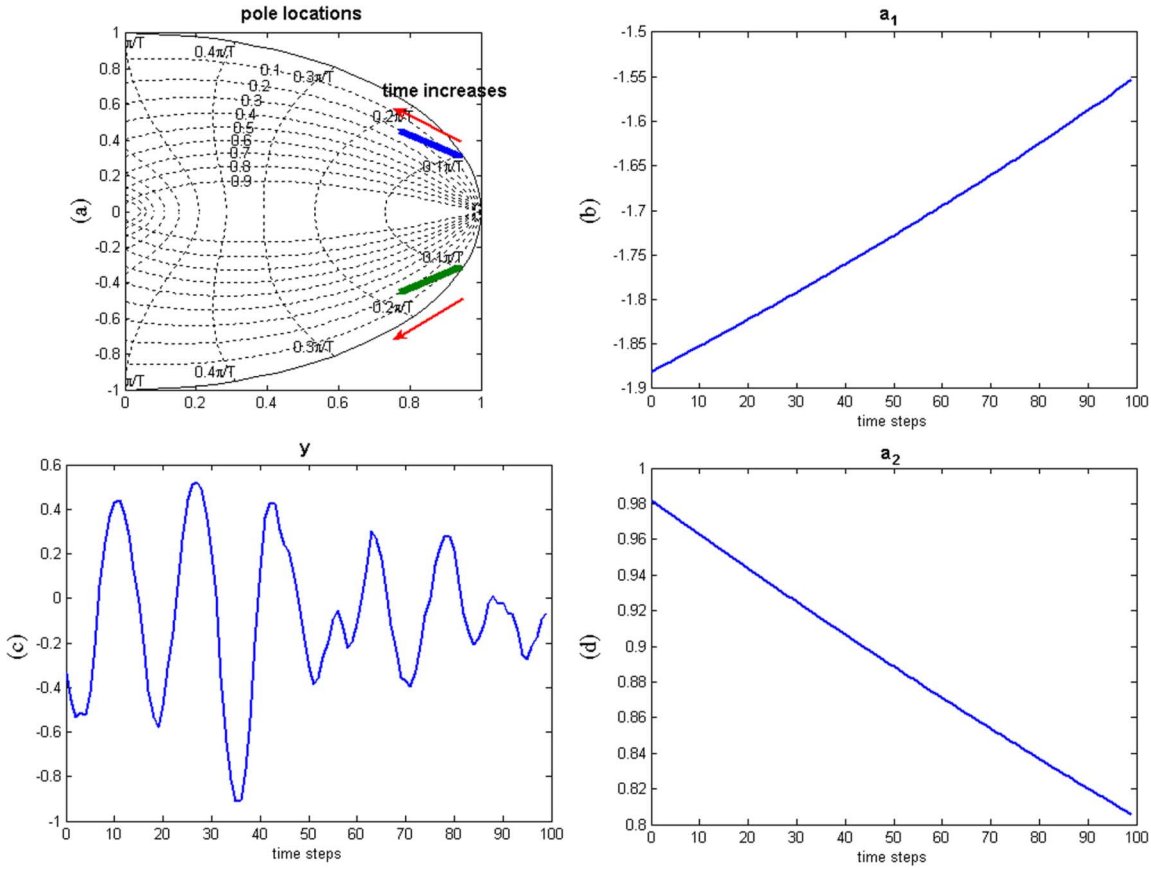


Fig. 1. Second-order AR system. (a) Pole locations. (b) $a_1(k)$. (c) $y(k)$. (d) $a_2(k)$.

where $\mathbf{e}_1 = [1, 0, \ldots, 0]^{\mathrm{T}} \in \mathbb{R}^N$. $\hat{\mathbf{a}}_{m,\mathrm{MAP}}$ and $\hat{a}_{m,\mathrm{MAP}}(k)$, $m = 1, \ldots, n$, denote the MAP estimates of $\mathbf{a}_m$ and $a_m(k)$, respectively.

Then, β_ε and β_i's are chosen to simplify (23). Let

$$\beta_\varepsilon = \rho - \frac{1}{2}\left(\mathbf{y} + \sum_{m=1}^{n} \mathbf{Y}_m \hat{\mathbf{a}}_{m,\mathrm{MAP}}\right)^T \left(\mathbf{y} + \sum_{m=1}^{n} \mathbf{Y}_m \hat{\mathbf{a}}_{m,\mathrm{MAP}}\right) \tag{24}$$

$$\beta_i = \frac{1}{K_i}\rho - \left(\frac{1}{2}\hat{\mathbf{a}}_{i,\mathrm{MAP}}^T \mathbf{D} \hat{\mathbf{a}}_{i,\mathrm{MAP}} - \hat{\mathbf{a}}_{i,\mathrm{MAP}}(0)\mu_i + \frac{1}{2}\mu_i^2\right), \quad i = 1, \ldots, n \tag{25}$$

where K_i is a positive real number which will be determined later and

$$\rho = \frac{1}{2}\left(\mathbf{y} + \sum_{m=1}^{n} \mathbf{Y}_m \hat{\mathbf{a}}_{m,\mathrm{MAP}}\right)^T \left(\mathbf{y} + \sum_{m=1}^{n} \mathbf{Y}_m \hat{\mathbf{a}}_{m,\mathrm{MAP}}\right) + \sum_{m=1}^{n}\left[\frac{1}{2}\hat{\mathbf{a}}_{m,\mathrm{MAP}}^T \mathbf{D} \hat{\mathbf{a}}_{m,\mathrm{MAP}} - \hat{\mathbf{a}}_{m,\mathrm{MAP}}(0)\mu_m + \frac{1}{2}\mu_m^2\right].$$

Define the parameter estimation error of $\mathbf{a}_i$ and $i = 1, \ldots, n$ to be

$$\tilde{\mathbf{a}}_i = \hat{\mathbf{a}}_{i,\mathrm{MAP}} - \mathbf{a}_i^*. \tag{26}$$

Substituting (24)–(26) into (23) and rearranging the equation, we end up with

$$\left(\mathbf{Y}_i^2 + K_i \mathbf{D}\right)\tilde{\mathbf{a}}_i + \mathbf{Y}_i \sum_{m=1, m\neq i}^{n} \mathbf{Y}_m \tilde{\mathbf{a}}_m = K_i \mu_i \mathbf{e}_1 - K_i \mathbf{D}\mathbf{a}_i^* - \mathbf{Y}_i \boldsymbol{\varepsilon}. \tag{27}$$

Combine (27) for all $i = 1, \ldots, n$ in a matrix form. Then

$$\begin{bmatrix} \tilde{\mathbf{a}}_0 \\ \tilde{\mathbf{a}}_1 \\ \vdots \\ \tilde{\mathbf{a}}_n \end{bmatrix} = \begin{bmatrix} \mathbf{Y}_1^2 + K_1\mathbf{D} & \mathbf{Y}_1\mathbf{Y}_2 & \cdots & \mathbf{Y}_1\mathbf{Y}_N \\ \mathbf{Y}_2\mathbf{Y}_1 & \mathbf{Y}_2^2 + K_2\mathbf{D} & \cdots & \mathbf{Y}_2\mathbf{Y}_N \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{Y}_N\mathbf{Y}_1 & \mathbf{Y}_N\mathbf{Y}_2 & \cdots & \mathbf{Y}_N^2 + K_N\mathbf{D} \end{bmatrix}^{-1} \times \begin{bmatrix} K_1\mu_1\mathbf{e}_1 - K_1\mathbf{D}\mathbf{a}_1^* - \mathbf{Y}_1\boldsymbol{\varepsilon} \\ K_2\mu_2\mathbf{e}_1 - K_2\mathbf{D}\mathbf{a}_2^* - \mathbf{Y}_2\boldsymbol{\varepsilon} \\ \vdots \\ K_n\mu_n\mathbf{e}_1 - K_n\mathbf{D}\mathbf{a}_n^* - \mathbf{Y}_n\boldsymbol{\varepsilon} \end{bmatrix}. \tag{28}$$

Equation (28) gives the error of the MAP estimate provided that β_ε and β_i's follow (24) and (25).

K_i's in (25) affecting the bias and variance of the estimated parameters. This is because large K_i leads to small β_i (25), which, in turn, introduces a "flat" distribution of σ_i^{-2} (5). Consequently, it is likely that σ_i is very small and, therefore, $a_i(k)$ is confined to a small neighborhood of $a_i(k-1)$. In other words, the estimated parameters have a smooth fluctuation but are not

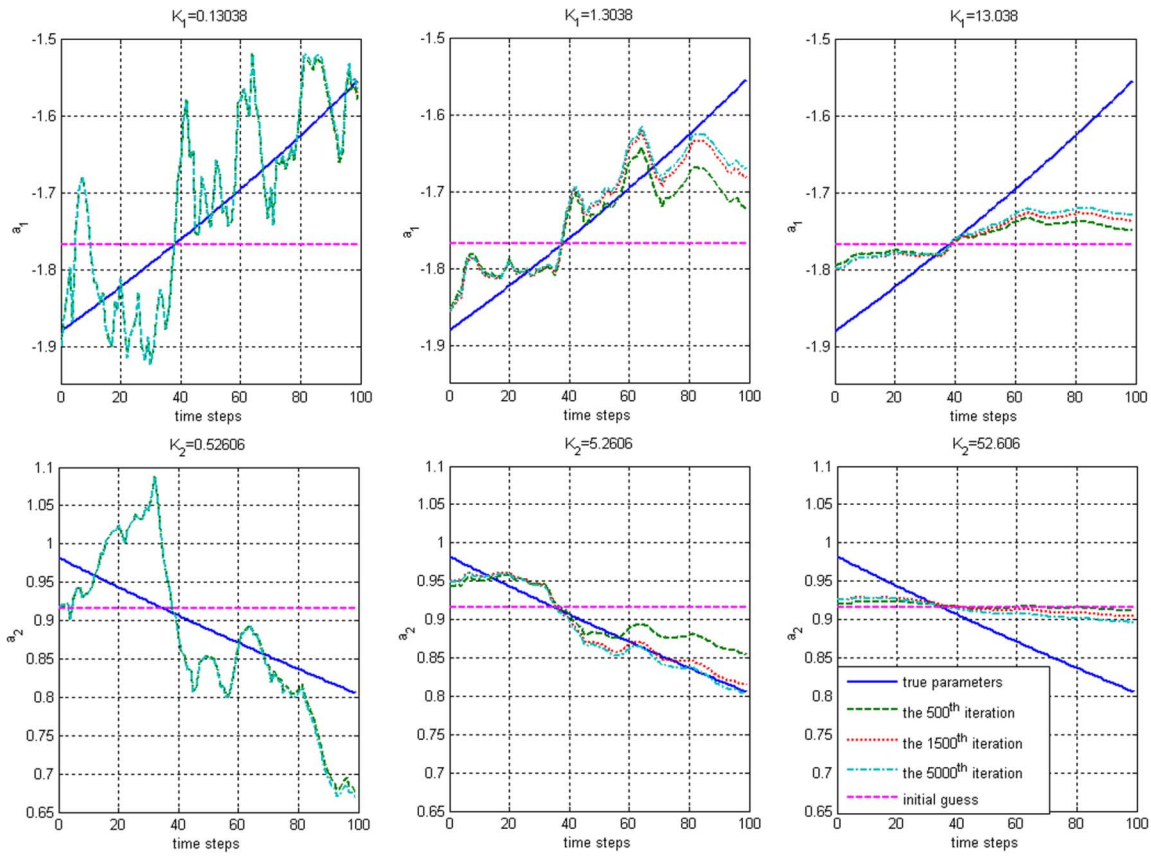


Fig. 2. Estimated parameters at various iterations. Left column $K_1 = 0.13038$ and $K_2 = 0.52606$. Middle column $K_1 = 1.3038$, $K_2 = 5.2606$. In the right column, $K_1 = 13.038$ and $K_2 = 52.606$. The solid line (-) indicates true parameters. The dashed line (--) indicates the 500th iteration. The dotted line (.) represents the 1500th iteration and the dashed–dotted line (-.) is the 5000th iteration.

able to trace large variations of the true parameters. Hence, the bias is large.

It is not easy to find "optimal" K_i's that achieve multiple conflicting goals, such as fast convergence, small bias, and small variance of estimation; however, (28) suggests the following rule of thumb in selecting K_i's. Namely choose K_i's such that the norm of the right-hand side of (28) is as small as possible. The minimal 2-norm of $K_i(\mu_i \mathbf{e}_1 - \mathbf{D}\mathbf{a}_i^*) - \mathbf{Y}_i\boldsymbol{\varepsilon}$ takes place whenever $K_i(\mu_i \mathbf{e}_1 - \mathbf{D}\mathbf{a}_i^*)$ is the orthogonal projection of $\mathbf{Y}_i\boldsymbol{\varepsilon}$ in the direction of $\mu_i \mathbf{e}_1 - \mathbf{D}\mathbf{a}_i^*$. Hence

$$K_i = \left| \frac{(\mu_i \mathbf{e}_1 - \mathbf{D}\mathbf{a}_i^*)^T \mathbf{Y}_i \boldsymbol{\varepsilon}}{(\mu_i \mathbf{e}_1 - \mathbf{D}\mathbf{a}_i^*)^T (\mu_i \mathbf{e}_1 - \mathbf{D}\mathbf{a}_i^*)} \right|. \tag{29}$$

Note that (29) is not an optimal choice of K_i in any sense. However, it is the author's experience that such a choice gives rise to satisfactory results. Simulations in the next section verify this point of view.

Equation (29) requires the values of true parameters $\mathbf{a}_i^*$ and the process noise $\boldsymbol{\varepsilon}$. Both of them are not available. Therefore, they can be replaced by their respective estimated values as follows:

$$\hat{K}_i^{(j)} = \left| \frac{\left(\mu_i \mathbf{e}_1 - \mathbf{D}\hat{\mathbf{a}}_i^{(j)}\right)^T \mathbf{Y}_i \hat{\boldsymbol{\varepsilon}}^{(j)}}{\left(\mu_i \mathbf{e}_1 - \mathbf{D}\hat{\mathbf{a}}_i^{(j)}\right)^T \left(\mu_i \mathbf{e}_1 - \mathbf{D}\hat{\mathbf{a}}_i^{(j)}\right)} \right|$$

where $\hat{\mathbf{a}}_i^{(j)}$ is the estimate of $\mathbf{a}_i^*$ at the jth iteration while $\hat{\boldsymbol{\varepsilon}}^{(j)} = \mathbf{y} + \sum_{i=1}^{n} \mathbf{Y}_i \hat{\mathbf{a}}_i^{(j)}$ is the estimate of $\boldsymbol{\varepsilon}$ at the jth iteration. However, it has been found that frequent updating of $\hat{K}_i^{(j)}$ may deteriorate the performance. In this case, $\hat{K}_i^{(j)}$ is updated every J iterations, where J is sufficiently large such that $\hat{\mathbf{a}}_i^{(j)}$ has converged within J iterations.

IV. Simulations

In order to investigate the strengths and weaknesses of the proposed method, it is desirable to compare the performance of Algorithm I with those of other methods. However, it is beyond the scope of this paper to conduct a comprehensive test of all kinds of time-varying system identification techniques. Simulations have been conducted to show that the RLS algorithm (with forgetting factors) results in unsatisfactory performance in the case of fast-varying systems while Algorithm I still works properly; however, the comparisons with the RLS algorithm are skipped in this paper due to the limited space. Instead, a particle filter [27] is implemented in this section as a comparison of Algorithm I. Particle filtering is chosen because it also belongs to the category of stochastic methods; hence, the comparison can be made on a fair basis. Both Algorithm I and the particle filter are implemented by C++ language and executed on the same personal computer (with a 3-GHz Pentium 4 CPU). The execution time and estimation errors of both methods are presented for comparison.

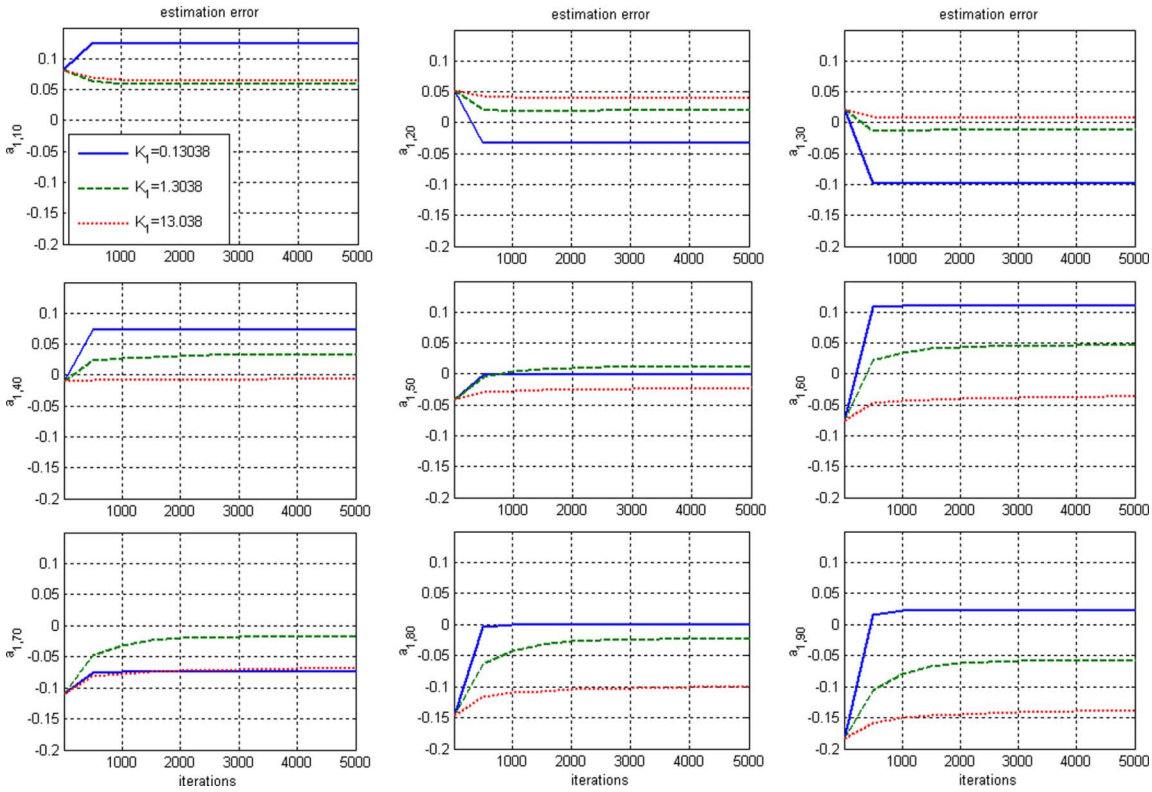


Fig. 3. Convergence of the estimation errors of $a_1(k)$ as the algorithm iterates for various K_1. The solid line (-) represents $K_1 = 0.13038$. The dashed line (--) is $K_1 = 1.3038$. The dotted line (.) is $K_1 = 13.038$.

In this section, we consider a time-varying second-order AR system

$$y(k) = -a_1(k)y(k-1) - a_2(k)y(k-2) + \varepsilon(k) \qquad (30)$$

where $\varepsilon(k)$ is a zero mean white Gaussian noise with standard deviation $\sigma_\varepsilon = 0.05$. Let the system's instantaneous poles be $p(k) = r(k)\mathrm{e}^{\pm j\theta(k)}$. Let $r(k) = 0.995 \cdot (0.999)^{k-1}$ and $\theta(k) = 0.1 \cdot \pi \cdot (1.005)^{k-1}$. Thus, $a_1(k) = -2r(k)\cos\theta(k)$ and $a_2(k) = r^2(k)$. Note that in this case, $a_1(k)$ and $a_2(k)$ are not independent, which violates the assumption of (4). However, the simulation results show that the Algorithm I still yields satisfactory estimation. The pole locations $p(k)$, parameters $a_1(k)$ and $a_2(k)$, and the output $\underline{y}(k)$ are shown in Fig. 1 for $k = 0, 1, \ldots, 99$.

If this system is regarded as time invariant, the least-square estimates are $\hat{a}_{1,\mathrm{LS}} = -1.7672$ and $\hat{a}_{2,LS} = 0.9158$. Clearly, the least-square estimates are not able to capture the time-varying features of the AR system. Nevertheless, they can serve as the initial guesses of Algorithm I and the particle filter.

Before implementing Algorithm I and the particle filter, we explain why the Markov Chain Monte Carlo (MCMC) method is not recommended in this case. A crucial step of the MCMC method is to draw samples of the unknown parameters in a delicate way such that the probability distribution of these samples asymptotically approaches the desired posterior probability. However, it is not an easy task to design an efficient sampler due to the complexity of (11)–(13). The Gibbs sampler [23] is widely used in the case that the dimension of the parameter space is high. Then, samples of each $a_i(k)$ are drawn based on the conditional probability proportional to $f_{i,k}g_{i,k}$ [(15) and (16)]. This can be achieved by drawing samples from $f_{i,k}$ (or $g_{i,k}$) directly and $g_{i,k}$ (or $f_{i,k}$) plays the role of the importance function [28]. However, as the length of the data increases, the high-probability regions of $f_{i,k}$ and $g_{i,k}$ diminish and do not overlap, which makes the importance sampling inefficient. Due to its low efficiency, the MCMC method will not be applied in this section.

A. Simulation Results of Algorithm I

For simplicity, fixed-value K_i's are used in this section. Three sets of K_1 and K_2 are selected to investigate their effects on estimation errors. According to (29), $K_1 = 1.3038$ and $K_2 = 5.2606$. The values of the other two sets of K_1 and K_2 are chosen to be the 10 and 1/10 times of the first set, respectively. Then, Algorithm I is applied to estimate $a_1(k)$ and $a_2(k)$ for $k = 0, 1, \ldots, 99$. The results are shown in Fig. 2. The estimation errors of $a_1(k)$ and $a_2(k)$ at nine selected time points for various iterations are shown in Figs. 3 and 4, respectively. These figures illustrate the convergence rate of Algorithm I.

It is clear from the simulation that larger K_1 and K_2 result in smaller variations among the estimated parameters and slower convergence rate. If K_1 and K_2 are too large, the estimated parameters are nearly constant for all k. This observation coincides with the discussion in Section III-B. It is also observed that the values of K_i's suggested by (29) yield satisfactory results in terms of bias and variance of the estimation (see the middle column of Fig. 2).

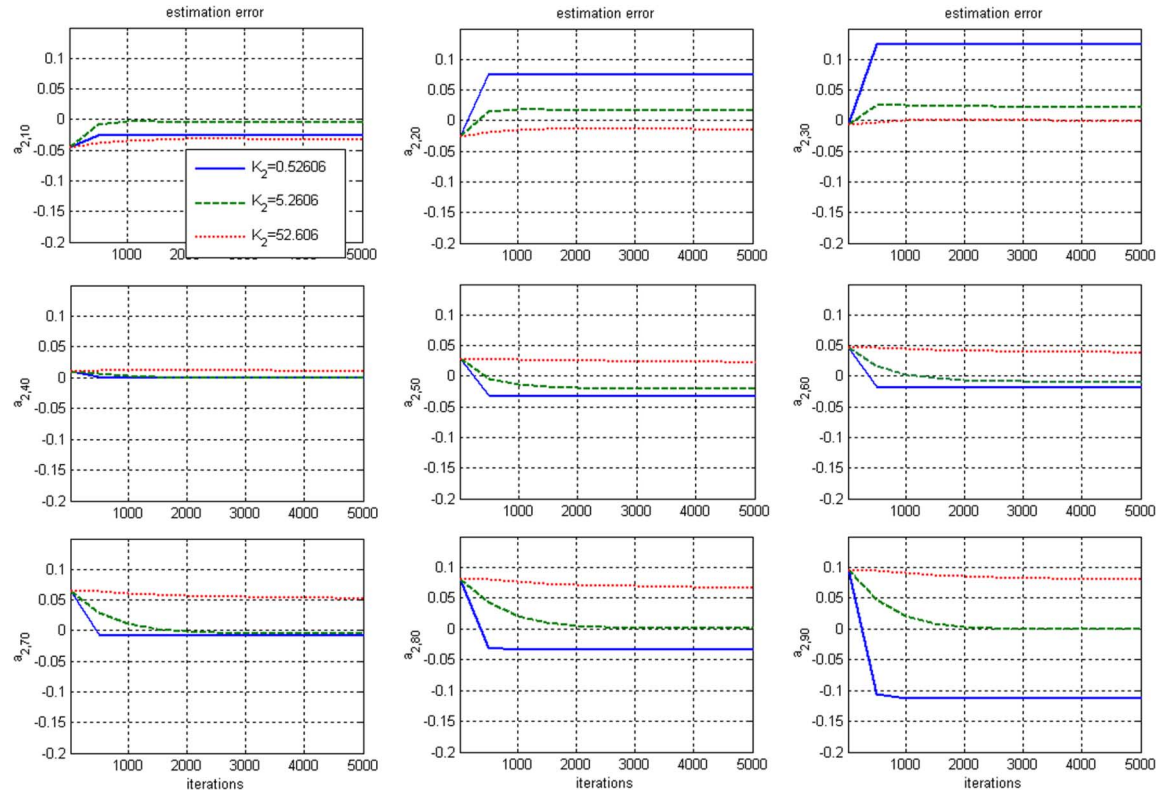


Fig. 4. Convergence of estimation errors of $a_2(k)$ as the algorithm iterates for various K_2. The solid line (-) indicates $K_2 = 0.52606$. The dashed line (--) is $K_2 = 5.2606$. The dotted line (.) is $K_2 = 52.606$.

TABLE I
SIMULATION RESULTS OF ALGORITHM I (AFTER 5000 ITERATIONS)

	K_1=0.13038 K_2=0.52606	K_1=1.3038 K_2=5.2606	K_1=13.038 K_2=52.606
Execution Time (sec)	2	2	2
Estimation Error of $\mathbf{a}_1$	7.4709×10^{-3}	4.3307×10^{-3}	7.5311×10^{-3}
Estimation Error of $\mathbf{a}_2$	7.3031×10^{-3}	1.4580×10^{-3}	4.6217×10^{-3}

TABLE II
SIMULATION RESULTS OF THE PARTICLE FILTER

Sample Size M	1,000	10,000	100,000
Average Execution Time for Each Run	<1 sec	8 sec	603 sec
Average Estimation Error of $\mathbf{a}_1$	1.2076×10^{-2}	1.0530×10^{-2}	9.8837×10^{-3}
Std* of the Estimation Error of $\mathbf{a}_1$	6.9756×10^{-3}	1.8717×10^{-3}	6.9412×10^{-4}
Average Estimation Error of $\mathbf{a}_2$	1.6054×10^{-2}	1.7860×10^{-2}	1.6638×10^{-2}
Std of the Estimation Error of $\mathbf{a}_2$	7.2390×10^{-3}	3.4232×10^{-3}	1.0008×10^{-3}

*: Std denotes "standard deviation"

Simulation results are listed in Table I. The estimation error is defined as $||\mathbf{a}_i^* - \hat{\mathbf{a}}_i||_2/N$, $i = 1, 2$. It can be seen that the K_i's suggested by (29) result in the smallest errors.

B. Simulation Results of the Particle Filter

The particle filter implemented in this section consists of the following steps.

Choose sample size M and hyperparameters α_i, β_i, i = 1, 2, α_ε and β_ε.

Given initial values of $\{a_i^{(m)}, w^{(m)}\}$, $i = 1, 2$, and $m = 1, \ldots, M$.

For each k, $k = 0, 1, \ldots, N-1$.

1) Draw samples of $\sigma_i^{-2} \sim Ga(\cdot|\alpha_i, \beta \mathrm{i}^{-1})$, $i = 1, 2$, and $\sigma_\varepsilon^{-2} \sim Ga(\cdot|\alpha_\varepsilon, \beta_\varepsilon^{-1})$.

2) Draw samples of $\overline{a}_i^{(m)} \sim p(\cdot|a_i^{(m)}, \sigma_i^{-2})$, $i = 1, 2$; $m = 1, \ldots, M$.

3) $\overline{w}^{(m)} = w^{(m)} p(y_k|\overline{a}_1^{(m)}, \overline{a}_2^{(m)}, \sigma_\varepsilon^{-2})$.

4) Apply residual resampling if the effective sample size is smaller than the prescribed threshold [29].

5) $\{a_i^{(m)}, w^{(m)}\} = \{\overline{a}_i^{(m)}, \overline{w}^{(m)} / \sum_{i=1}^{M} \overline{w}^{(i)}\}$, $i = 1, 2$, and $m = 1, \ldots, M$.

6) The estimated parameter at time step k is $\hat{a}_i(k) = \sum_{m=1}^{M} w^{(m)} a_i^{(m)}$, $i = 1, 2$.

The values of the hyperparameters are set to be the same as those in the Algorithm I (i.e., $\alpha_1 = \alpha_2 = \alpha_\varepsilon = 1, \beta_1 = 0.4767$, $\beta_2 = 0.1181$, and $\beta_\varepsilon = 0.5007$ corresponding to $K_1 = 1.3038$ and $K_2 = 5.2606$) [(24) and (25)]. Note that uncertainty exists in the estimates of the particle filter. Hence, for each sample size M, we run the particle filter ten times. The average execution time for each run, and the mean and the standard deviation of

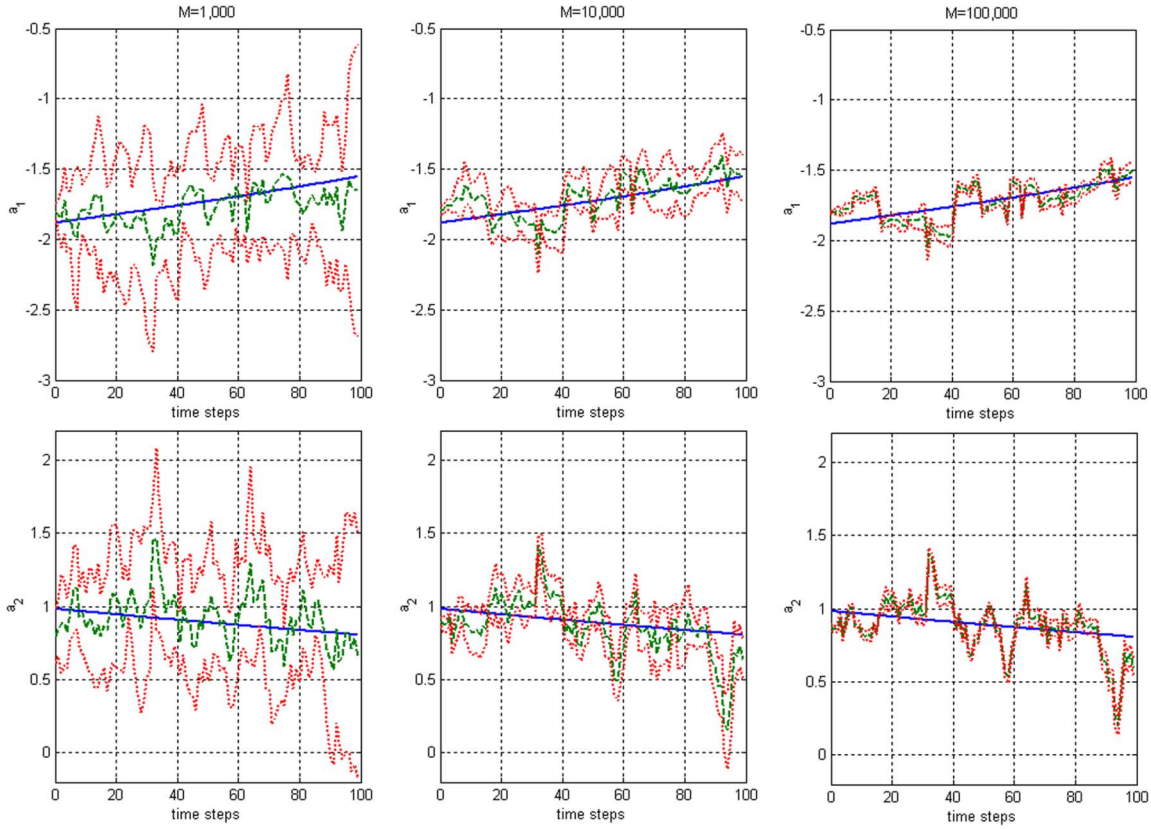


Fig. 5. Estimated parameters for various sample sizes. Left column $\mathrm{M} = 1,000$. Middle column $\mathrm{M} = 10,000$. Right column $\mathrm{M} = 100,000$. Upper row a_1. Lower row a_2. The solid line (-) represents true parameters. The dashed line (--) indicated estimated parameters. The dotted line (.) is the standard deviation of the estimated parameters.

the estimation errors are listed in Table II. Fig. 5 shows the true and estimated parameters.

C. Discussion

It can be seen from Table II that an increase of the sample size reduces the variances of the estimation errors, but has minor effects on the average estimation errors. Comparing the results in Tables I and II, we found that Algorithm I achieves smaller estimation errors with less computation power. Moreover, the result of Algorithm I is deterministic (i.e., the same MAP estimate is obtained for every run provided that the same hyperparameters are used). On the other hand, the particle filter can perform online estimation of the parameters but the result is uncertain. A large sample size reduces the uncertainties; however, more computation power is required.

V. Conclusion

In this paper, the problem of identifying time-varying parameters of autoregressive (AR) systems was investigated. It was formulated as a Bayesian inference problem with additional constraints imposed on the parameters in the form of conditional and prior probabilities. These constraints represent the objective knowledge or the subjective belief about the physical system; however, the conditional Gaussian distributions and the corresponding conjugate priors arise naturally if little about the system is revealed.

Based on the probabilistic assumptions, an efficient iterative algorithm was proposed to evaluate the maximum *a posteriori* (MAP) estimates of parameters. Simulation results showed that the proposed method has satisfactory performance. Compared with the particle filter, the proposed method achieved smaller estimation errors with less computation power.

In this paper, the order of the AR system is assumed to be known in advance. In reality, this information is usually not available. Determining the order of the identified system is called "model selection" in system identification literature. Criteria, such as Akaike's Information Criterion (AIC), Bayesian Information Criterion (BIC), and final prediction error (FPE), have been proposed for linear time-invariant systems [5], but the model selection for time-varying systems remains an open question. Although the order of the time-varying system can be roughly estimated by the aforementioned criteria, assuming temporarily that the system is time invariant, there is no guarantee that the correct model will be selected. Model selection for time-varying systems will be a future research topic.

In addition to the AR model, there are many other parametric representations of systems, such as ARX and ARMAX models [5]. In these cases, the "amount of information" contained in the input signal becomes an issue. If the information is not "rich" enough, only parts of the system's characteristics will be activated. However, a rigorous presentation of the identifiability of the time-varying system and the relation to the properties of the input–output signals requires more research effort in the future.

REFERENCES

[1] Y. Grenier, "Time-dependent ARMA modeling of nonstationary signals," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-31, no. 4, pp. 899–911, Aug. 1983.

[2] R. A. Iltis and A. W. Fuxjaeger, "A digital spread-spectrum receiver with joint channel and Doppler shift estimation," *IEEE Trans. Commun.*, vol. 39, no. 8, pp. 1255–1267, Aug. 1991.

[3] A. Singhal and A. S. Kiremidjian, "Method for probabilistic evaluation of seismic structural damage," *J. Struct. Eng.*, vol. 122, pp. 1459–1467, 1996.

[4] J. P. Kaipio and P. A. Karjalainen, "Estimation of event-related synchronization changes by a new TVAR method," *IEEE Trans. Biomed. Eng.*, vol. 44, no. 8, pp. 649–656, Aug. 1997.

[5] L. Ljung, *System Identification: Theory for the User*, 2nd ed. Upper Saddle River, NJ: Prentice-Hall, 1999.

[6] R. Johansson, *System Modeling and Identification*. Englewood Cliffs, NJ: Prentice-Hall, 1993.

[7] P. A. Ioannou and J. Sun, *Robust Adaptive Control*. Upper Saddle River, NJ: Prentice-Hall, 1996.

[8] K. Nishiyama, "An H^{∞} optimization and its fast algorithm for time-variant system identification," *IEEE Trans. Signal Process.*, vol. 52, no. 5, pp. 1335–1342, May 2004.

[9] J. C. M. Bermudez and N. I. Bershad, "Transient and tracking performance analysis of the quantized lms algorithm for time-varying system identification," *IEEE Trans. Signal Process.*, vol. 44, no. 8, pp. 1990–1997, Aug. 1996.

[10] R. Zou and K. H. Chon, "Robust algorithm for estimation of time-varying transfer functions," *IEEE Trans. Biomed. Eng.*, vol. 51, no. 2, pp. 219–228, Feb. 2004.

[11] M. K. Tsatsanis and G. B. Giannakis, "Time-varying system identification and model validation using wavelets," *IEEE Trans. Signal Process.*, vol. 41, no. 12, pp. 3512–3523, Dec. 1993.

[12] Y. Zheng, D. B. H. Tay, and Z. Lin, "Modeling general distributed nonstationary process and identifying time-varying autoregressive system by wavelets: Theory and application," *Signal Process.*, vol. 81, pp. 1823–1848, 2001.

[13] K. B. Eom, "Analysis of acoustic signatures from moving vehicles using time-varying autoregressive models," *Multidimensional Syst. Signal Process.*, vol. 10, pp. 357–378, 1999.

[14] K. B. Eom, "Nonstationary autoregressive contour modeling approach for planar shape analysis," *Opt. Eng.*, vol. 38, pp. 1826–1835, 1999.

[15] G. N. Fouskitakis and S. D. Fassois, "On the estimation of nonstationary functional series TARMA models: An isomorphic matrix algebra based method," *J. Dyn. Syst., Meas. Control*, vol. 123, pp. 601–610, 2001.

[16] M. K. Tsatsanis and G. B. Giannakis, "Subspace methods for blind estimation of time-varying FIR channels," *IEEE Trans. Signal Process.*, vol. 45, no. 12, pp. 3084–3093, Dec. 1997.

[17] K. B. Eom, "Time-varying autoregressive modeling of high range resolution radar signatures for classification of noncooperative targets," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 35, no. 3, pp. 974–988, Jul. 1999.

[18] A. Doucet, S. J. Godsill, and M. West, "Monte Carlo filtering and smoothing with application to time-varying spectral estimation," presented at the IEEE Int. Conf. Acoustics, Speech Signal Processing II, Istanbul, Turkey, 2000.

[19] S. J. Godsill, A. Doucet, and M. West, "Monte Carlo smoothing for nonlinear time series," *J. Amer. Statist. Assoc.*, vol. 99, pp. 156–168, 2004.

[20] S. Godsill and T. Clapp, "Improvement strategies for Monte Carlo particle filters," in *Sequential Monte Carlo Methods in Practice*, A. Doucet, N. D. Freitas, and N. Gordon, Eds. Berlin, Germany: Springer, 2001, pp. 139–158.

[21] J. M. Bravo, T. Alamo, and E. F. Camacho, "Bounded error identification of systems with time-varying parameters," *IEEE Trans. Autom. Control*, vol. 51, no. 7, pp. 1144–1150, Jul. 2006.

[22] V. Gemez and A. Maravall, "Estimation, prediction, and interpolation for nonstationary series with the Kalman filter," *J. Amer. Stat. Assoc.*, vol. 89, pp. 611–624, 1994.

[23] C. P. Robert and G. Casella, *Monte Carlo Statistical Methods*. Berlin, Germany: Springer, 2004.

[24] J. M. Bernardo and A. F. M. Smith, *Bayesian Theory*. New York: Wiley, 2000.

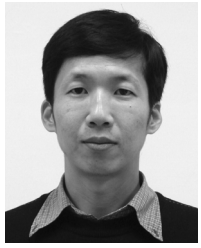
[25] R. E. Neapolitan, *Learning Bayesian Networks*. Upper Saddle River, NJ: Prentice-Hall, 2004.

[26] M. Vidyasagar, *Nonlinear Systems Analysis*, 2nd ed. Englewood Cliffs, NJ: Prentice-Hall, 1993.

[27] S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for on-line non-linear/non-Gaussian Bayesian tracking," *IEEE Trans. Signal Proces.*, vol. 50, no. 2, pp. 174–188, Feb. 2002.

[28] C. Andrieu, N. D. Freitas, A. Doucet, and M. I. Jordan, "An introduction to MCMC for machine learning," *Mach. Learn.*, vol. 50, pp. 5–43, 2003.

[29] J. S. Liu and R. Chen, "Sequential Monte Carlo methods for dynamic systems," *J. Amer. Stat. Assoc.*, vol. 93, pp. 1032–1044, 1998.



Tesheng Hsiao (M'05) received the B.S. and M.S. degrees in control engineering from National Chiao Tung University, Hsinchu, Taiwan, R.O.C., in 1995 and 1997, respectively, and the Ph.D. degree in mechanical engineering from the University of California, Berkeley, in 2005.

Currently, he is an Assistant Professor in the Department of Electrical and Control Engineering at National Chiao Tung University. His research interests include advanced vehicle-control systems, fault detection and fault-tolerant control, and system identification.