

Parameter Estimation of RBF-AR Model Based on the EM-EKF Algorithm

Yanhui Xi^{1,2} Hui Peng³ Hong Mo¹

Abstract RBF-AR (radial basis function network-based autoregressive) model is reconstructed as a new type of general radial basis function (RBF) neural network, which has additional linear output weight layer in comparison with the traditional three-layer RBF network. The extended Kalman filter (EKF) algorithm for RBF training has low filtering accuracy and divergence because of unknown prior knowledge, such as noise covariance and initial states. To overcome the drawback, the expectation maximization (EM) algorithm is used to estimate the covariance matrices of noises and the initial states. The proposed method, called the EM-EKF (expectation-maximization extended Kalman filter) algorithm, which combines the expectation maximization, extended Kalman filtering and smoothing process, is developed to estimate the parameters of the RBF-AR model, the initial conditions and the noise variances simultaneously. It is shown by the simulation tests that the EM-EKF method for the reconstructed RBF-AR network provides better results than structured nonlinear parameter optimization method (SNPOM) and the EKF, especially in low SNR (signal noise ratio). Moreover, the EM-EKF method can accurately estimate the noise variance. F test indicates there is significant difference between results obtained by the SNPOM and the EM-EKF.

Key words Expectation maximization (EM) algorithm, extended Kalman filtering (EKF) and smoothing, radial basis function (RBF) neural network, RBF-AR model

Citation Yanhui Xi, Hui Peng, Hong Mo. Parameter estimation of RBF-AR model based on the EM-EKF algorithm. *Acta Automatica Sinica*, 2017, 43(9): 1636–1643

DOI 10.16383/j.aas.2017.e160216

1 Introduction

The radial basis function network-based state-dependent autoregressive (RBF-AR) model, which offers a very flexible structure for nonlinear time-series modeling, has been extensively studied. For example, Shi *et al.* [1] developed the RBF-AR model to reconstruct the dynamics of given nonlinear time series. Peng *et al.* [2] extended the RBF-AR model to the case where there are several exogenous variables (RBF-ARX model) for the system. Following this method, Gan *et al.* [3], [4] successively developed the locally linear radial basis function network-based autoregressive (LLRBF-AR) model and a gradient radial basis function based varying-coefficient autoregressive (GRBF-AR) model. The major feature of the RBF-AR (X) model which is superior to the black-box models based on general function approximations is that: the RBF-AR (X) model may provide some insights into the system dynamics due to its quasi-linear structure, whereas the general function approximations cannot.

The identification of the RBF networks includes the choice of topology (e.g., the number of neurons) and estimation of all the parameters. For selecting the number of neurons, several methods have been proposed, e.g., the Akaike information criterion (AIC), Bayesian information criterion (BIC), and cross validation. Therefore we must first have a good model parameter estimation method, and then we can repeat the method for models with different number of neurons, before finally selecting the best model. In this paper, the majority of our works focuses on deter-

mining the parameters of RBF networks.

The estimation of the RBF-AR (X) model is a difficult task but is a key procedure for successfully applying the RBF-AR (X) model. Peng *et al.* [2] presented a structured nonlinear parameter optimization method (SNPOM) for the RBF-AR (X) model estimation. Researches [2], [5], [6] shown the SNPOM can optimize all of the model parameters and can also accelerate the computational convergence of the optimization search process. However, the primary shortcoming of the SNPOM is that it can be easily trapped in a local optimum because of an initial guess value. To overcome the problem of local minima, Gan *et al.* [7] proposed the hybrid evolutionary algorithm (EA)-SNPOM and simulation results indicated that the hybrid algorithm provides better results than either method alone (EA and SNPOM), but the complexity and running time increased substantially. Also, Gan *et al.* [8] proposed a variable projection (VP) approach to efficiently estimate the parameters of the RBF-ARX model. Although these identification methods for RBF-AR (X) have good estimation results, their performance may get deteriorated in case of measurement with noise interference.

As Peng *et al.* [2] and Gan *et al.* [7] mentioned that RBF-AR (X) can be regarded as a more general nonlinear model than the RBF neural network and as a generalization of the RBF network. Furthermore, any kind of RBF and the RBF-ARX model parameter estimation procedure must include the selection of appropriate centers and scaling factors, and estimation of all the linear weights of the RBF networks in the model.

Various derivative-based methods have been used to train RBF neural network, including gradient descent [9] and the well-known back propagation [10]. Although gradient descent training for RBF networks has proven to be much more effective than lots of conventional methods, it can be computationally expensive. An alternative to sequentially estimate RBF network is the extended Kalman filter (EKF)-based neural network sequential learning method [11]–[13]. The sequential EKF algorithm is a fast training algorithm, which takes all the network parameters (weights or weights and centers) as a

Manuscript received November 14, 2016; accepted December 20, 2016.

This work was supported by the National Natural Science Foundation of China (51507015, 61773402, 61540037, 71271215, 61233008, 51425701, 70921001, 51577014), the Natural Science Foundation of Hunan Province (2015JJ3008), the Key Laboratory of Renewable Energy Electric-Technology of Hunan Province (2014ZNDL002), and Hunan Province Science and Technology Program (2015NK3035).

Recommended by Associate Editor Zhanshan Wang.
1. School of Electrical and Information Engineering, Changsha University of Science and Technology, Changsha 410076, China 2. School of Information Systems and Management, National University of Defense Technology, Changsha 410073, China 3. School of Information Science and Engineering, Central South University, Changsha 410083, China

state vector in state-space and adjusts them to adapt to measurements [14]. Because it makes use of second order statistics (covariance), the EKF is an improvement over conventional neural network estimation techniques, such as back-propagation and gradient descent training algorithm. However, a well known limitation of the EKF is the assumption of known a priori statistics to describe the measurement and process noise. Setting these noise levels appropriately often makes the difference between success and failure in the use of the EKF. Especially in environments where the noise statistics change with time, the EKF can lead to large estimation errors and even to divergence. Thus, the selection of the initial states, the measurement noise covariance matrix and the process noise covariance matrix are very important in the control of the convergence of the EKF learning algorithm. To overcome the difficulties above, a number of improved learning methods were proposed, such as EKFQ (the EKF algorithm with evidence maximization and sequentially updated priors) [15], [16], APNCPF (adaptive process noise covariance particle filter) [17], a hybrid EKF and switching PSO (particle swarm optimization) algorithm [18], and a particle filtering algorithm in combination with the kernel smoothing method [19]. The EKFQ and APNCPF algorithms employ adaptive noise parameters, but the problem of choosing the right window length, which is used to update noise covariance matrices, results in a regularization/tracking dilemma because of unknown prior knowledge. Moreover, the above algorithms do not consider the optimal initial values of parameters, which will lead to some imprecise information about the spread or the shape of the posterior [20].

The expectation-maximization (EM) algorithm [21] was developed to learn parameters of statistical models in the presence of incomplete data or hidden variables. Lázaro *et al.* [22] and Constantinopoulos *et al.* [23] extended the EM learning strategy for estimation of the neural network weights, such as the multi-layer perception (MLP) and the Gaussian radial basis function (RBF) networks. Research results indicate that the EM method is effective and simple to obtain maximum likelihood (ML) estimates of the parameter and states.

In this paper, the proposed expectation-maximization extended Kalman filter (EM-EKF) method, which combines the expectation-maximization and the extended Kalman filtering, is developed to identify the RBF-AR model. The method is more accurate as it involves extended Kalman smoothing, which provides a minimum variance Gaussian approximation to the posterior probability density function. To use the EM-EKF algorithm for state space learning, RBF-AR model is reconstructed as a general RBF network, which has additional linear output weight layer compared with the traditional three-layer RBF network. Thus, the general RBF network is represented by state-space model, and the centers and the weights of the general RBF network are treated as hidden state variables. The learning algorithm for RBF-AR model possesses advantages of both the expectation-maximization and of the extended Kalman filtering and smoothing. Specifically, the EM algorithm is utilized to estimate parameters of the state-space model, such as the measurement and process noise variance and the initial states, while the extended Kalman filtering and smoothing are used to estimate the approximate state distribution. The proposed algorithm simplifies the optimizing estimation of the maximum likelihood by making the expectation maximal, and EKF and smoothing process realize the more exact estimation of the expectation. This learning technique can improve the

performance of the EKF-based neural network sequential learning method by estimating the noise variance and the initial states. The performance and effectiveness of our proposed method are evaluated by the Mackey-Glass time series through three cases.

The contributions of this paper comprise two parts. First, RBF-AR model is reconstructed as a new type of general radial basis function (RBF) neural network, which makes it able to estimate the parameters of RBF-AR model using the EKF. Second, by combining the expectation maximization and extended Kalman filtering and smoothing process, the EM-EKF method is developed to estimate RBF-AR model, which can give joint state and parameter estimates.

The structure of the remaining of this paper is as follows. Section 2 presents the state-space representation of the RBF-AR model. The EM-EKF method is developed to identify the RBF-AR model in Section 3. The cases studies are shown in Section 4. Finally, the paper is concluded in Section 5.

2 The State-space Representation of the RBF-AR Model

We are interested in the nonlinear time series that can be described by the following state-dependent AR (SD-AR) model

$$\begin{cases} y_t = \phi_0(\mathbf{X}_{t-1}) + \sum_{i=1}^p \phi_i(\mathbf{X}_{t-1})y_{t-i} + e_t \\ \mathbf{X}_{t-1} = [y_{t-1}, y_{t-2}, \dots, y_{t-d}]^T \end{cases} \quad (1)$$

where y_t ($t = 1, \dots, T$) is the output, \mathbf{X}_{t-1} is regarded as the state vector at time t , which contains only the output series in this case (in other cases it may contain the input series or another). $\phi_i(\mathbf{X}_{t-1})$ ($i = 0, 1, \dots, p$) are the state-dependent function coefficients of the model, p is the model order, and d is the dimension of the state vector. e_t denotes Gaussian white noise.

Although the SD-AR model provides a useful framework for general nonlinear time series modeling, the problem is how to specify the functional form of its state-dependent coefficients. An efficient approach to solve the problem is to use Gaussian RBF neural networks approximations of coefficients of model (1) [2], and then the model derived is called the RBF-AR model, which is given by

$$\begin{cases} y_t = \phi_0(\mathbf{X}_{t-1}) + \sum_{i=1}^p \phi_i(\mathbf{X}_{t-1})y_{t-i} + e_t \\ \phi_0(\mathbf{X}_{t-1}) = \omega_{0,0} + \sum_{k=1}^m \omega_{0,k} \exp\{-\lambda_k \|\mathbf{X}_{t-1} - \mathbf{Z}_k\|_2^2\} \\ \phi_i(\mathbf{X}_{t-1}) = \omega_{i,0} + \sum_{k=1}^m \omega_{i,k} \exp\{-\lambda_k \|\mathbf{X}_{t-1} - \mathbf{Z}_k\|_2^2\} \\ \mathbf{Z}_k = [z_{k,1}, \dots, z_{k,d}]^T \end{cases} \quad (2)$$

where \mathbf{Z}_k ($k = 1, 2, \dots, m$) are the centers of the local linear RBF networks, λ_k ($k = 1, 2, \dots, m$) are the scaling parameters, $\omega_{i,k}$ ($i = 0, 1, 2, \dots, p; k = 0, 1, 2, \dots, m$) are the linear weights, m and d are the number of hidden neurons and the dimension of the centers (the dimension of the centers is the same as the dimension of the state vector), respectively, and $\|\cdot\|_2$ denotes the vector 2-norm.

In general case, the RBF networks in model (2) may have different centers for different regression variables. However, in some applications, all the RBF networks may be allowed to share the same centers, because model (2) possesses the

autoregressive structure, thus, although the RBF centers are the same in that case, the regressive polynomials' coefficients are different. Thus, the RBF-AR model can be seen as a general RBF network, which has two hidden layers with m and $p + 1$ neurons, respectively, and the output layer with one output. In this structure, the identification of RBF-AR model is to estimate the centers, the scaling parameters and the weights of the general RBF network. Fig. 1 shows the schematic of the RBF-AR model as a general RBF network.

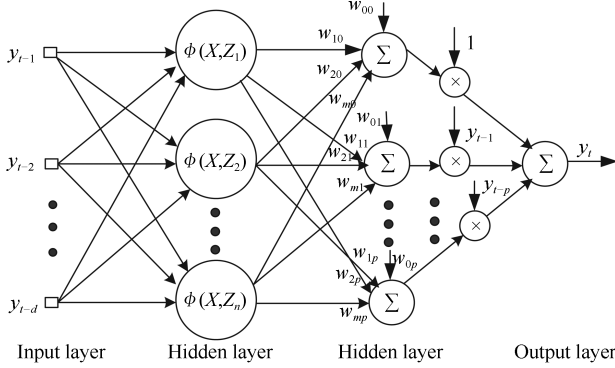


Fig. 1. The schematic of the RBF-AR model.

To simplify the nonlinear optimization steps, the scaling parameters may be selected as [2]

$$\lambda_k = \frac{-\log \epsilon_k}{\max_{t-1} \{\|\mathbf{X}_{t-1} - \mathbf{Z}_k\|^2\}}, \quad \epsilon_k \in [0.0001, 0.1]. \quad (3)$$

Using this heuristic way, the weights will become bounded and stable when the signal \mathbf{X}_{t-1} is far away from the centers \mathbf{Z}_k .

To apply a filtering algorithm to the RBF network training, the state-space model is established, which is given by

$$\begin{cases} \boldsymbol{\theta}_t = \boldsymbol{\theta}_{t-1} + \mathbf{v}_t \\ y_t = g(\boldsymbol{\theta}_t, \mathbf{X}_{t-1}) + \xi_t \\ = \phi_0(\boldsymbol{\theta}_t, \mathbf{X}_{t-1}) + \sum_{i=1}^p \phi_i(\boldsymbol{\theta}_t, \mathbf{X}_{t-1})y_{t-i} + \xi_t \end{cases} \quad (4)$$

where $\boldsymbol{\theta} = [\omega_0^T \ \omega_1^T \ \dots \ \omega_p^T \ \mathbf{Z}_1^T \ \dots \ \mathbf{Z}_m^T]^T$ ($\omega_j^T = [\omega_{0,j} \ \omega_{1,j} \ \dots \ \omega_{m,j}]$, $0 \leq j \leq p$; $\mathbf{Z}_k = [z_{k,1} \ z_{k,2} \ \dots \ z_{k,d}]^T$, $1 \leq k \leq m$) represents the system state vector, y_t is the observation variable, \mathbf{v}_t and ξ_t denote the process noise and observation noise, which are assumed to be zero-mean Gaussian processes with the covariance \mathbf{Q} and R , namely $\mathbf{v}_t \sim \mathcal{N}(0, \mathbf{Q})$, $\xi_t \sim \mathcal{N}(0, R)$. The initial state (parameters) $\boldsymbol{\theta}_0$ is normally-distributed with mean $\boldsymbol{\mu}_0$ and covariance $\boldsymbol{\Xi}_0$. Obviously, the crux of the matter is that both the system hidden state $\boldsymbol{\theta}_t$ and the parameters $\boldsymbol{\varphi} = (\mathbf{Q}, R, \boldsymbol{\mu}_0, \boldsymbol{\Xi}_0)$ are unknown.

3 The EM-EKF Method

To simultaneously estimate parameters and hidden states, the expectation maximization (EM) is incorporated with extended Kalman filtering and smoothing, which aims to integrate over the uncertain estimates of the unknown hidden states and optimize the resulting marginal likelihood of the parameters given the observed data. The algorithm realizes the more exact estimation of the posterior distribution by use of the extended Kalman filtering and smoothing.

The EM algorithm is an iterative method for finding a mode of the likelihood function. To derive the EM algorithm for nonlinear state space models, we need to develop an expression for the likelihood of the completed data. We assume that the likelihood of the data given the states, the initial conditions and the evolution of the states can be represented by Gaussian distributions. Thus, if the initial mean and covariance of the states is given by $\boldsymbol{\mu}_0$ and $\boldsymbol{\Xi}_0$, then

$$p(\boldsymbol{\theta}_0 | \boldsymbol{\varphi}) = (2\pi)^{-\frac{l}{2}} |\boldsymbol{\Xi}_0|^{-\frac{1}{2}} \times \exp \left[-\frac{1}{2} (\boldsymbol{\theta}_0 - \boldsymbol{\mu}_0)^T \boldsymbol{\Xi}_0^{-1} (\boldsymbol{\theta}_0 - \boldsymbol{\mu}_0) \right] \quad (5)$$

$$p(\boldsymbol{\theta}_t | \boldsymbol{\theta}_{t-1}, \boldsymbol{\varphi}) = (2\pi)^{-\frac{l}{2}} |\mathbf{Q}|^{-\frac{1}{2}} \times \exp \left[-\frac{1}{2} (\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t-1})^T \mathbf{Q}^{-1} (\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t-1}) \right] \quad (6)$$

$$p(y_t | \boldsymbol{\theta}_t, \boldsymbol{\varphi}) = (2\pi)^{-\frac{n}{2}} |R|^{-\frac{1}{2}} \times \exp \left[-\frac{1}{2} (y_t - g(\boldsymbol{\theta}_t, \mathbf{X}_{t-1}))^T R^{-1} (y_t - g(\boldsymbol{\theta}_t, \mathbf{X}_{t-1})) \right] \quad (7)$$

where $l = (1+m)(1+p) + md$ is the dimension of the state vector, $n = 1$ is the dimension of the observation vector.

Then, the log-likelihood of the complete data is given by:

$$\begin{aligned} \ln p(\boldsymbol{\theta}_{0:T}, y_{1:T} | \boldsymbol{\varphi}) &= -\frac{1}{2} [\boldsymbol{\theta}_0 - \boldsymbol{\mu}_0]^T \boldsymbol{\Xi}_0^{-1} [\boldsymbol{\theta}_0 - \boldsymbol{\mu}_0] \\ &\quad - \frac{1}{2} \ln |\boldsymbol{\Xi}_0| - \frac{T(l+n) + l}{2} \ln 2\pi - \frac{T}{2} \ln |\mathbf{Q}| \\ &\quad - \frac{T}{2} \ln |R| - \sum_{t=1}^T \frac{1}{2} [\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t-1}]^T \mathbf{Q}^{-1} [\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t-1}] \\ &\quad - \sum_{t=1}^T \frac{1}{2} [y_t - g(\boldsymbol{\theta}_t, \mathbf{X}_{t-1})]^T R^{-1} [y_t - g(\boldsymbol{\theta}_t, \mathbf{X}_{t-1})]. \end{aligned} \quad (8)$$

If we take the expectation of the log-likelihood for the complete data, we get the following expression:

$$\begin{aligned} E \ln [p(\boldsymbol{\theta}_{0:T}, y_{1:T} | \boldsymbol{\varphi})] &= -\frac{1}{2} \ln |\boldsymbol{\Xi}_0| - \frac{T}{2} \ln |\mathbf{Q}| - \frac{T}{2} \ln |R| \\ &\quad - \frac{1}{2} E \left[\boldsymbol{\theta}_0^T \boldsymbol{\Xi}_0^{-1} \boldsymbol{\theta}_0 - \boldsymbol{\theta}_0^T \boldsymbol{\Xi}_0^{-1} \boldsymbol{\mu}_0 - \boldsymbol{\mu}_0^T \boldsymbol{\Xi}_0^{-1} \boldsymbol{\theta}_0 + \boldsymbol{\mu}_0^T \boldsymbol{\Xi}_0^{-1} \boldsymbol{\mu}_0 \right] \\ &\quad - \frac{1}{2} \sum_{t=1}^T E \left[\boldsymbol{\theta}_t^T \mathbf{Q}^{-1} \boldsymbol{\theta}_t - \boldsymbol{\theta}_t^T \mathbf{Q}^{-1} \boldsymbol{\theta}_{t-1} - \boldsymbol{\theta}_{t-1}^T \mathbf{Q}^{-1} \boldsymbol{\theta}_t \right. \\ &\quad \left. + \boldsymbol{\theta}_{t-1}^T \mathbf{Q}^{-1} \boldsymbol{\theta}_{t-1} \right] - \frac{T(l+n) + l}{2} \ln 2\pi \\ &\quad - \frac{1}{2} \sum_{t=1}^T E \left[(y_t - g(\boldsymbol{\theta}_t, \mathbf{X}_{t-1}))^T R^{-1} (y_t - g(\boldsymbol{\theta}_t, \mathbf{X}_{t-1})) \right]. \end{aligned} \quad (9)$$

To compute the expectation of the measurement mapping $g(\boldsymbol{\theta}_t, \mathbf{X}_{t-1})$, the Taylor expansion of this mapping around the smoothing estimate $\boldsymbol{\theta}_{t|T}$ is given by

$$g(\boldsymbol{\theta}_t, \mathbf{X}_{t-1}) = g(\boldsymbol{\theta}_{t|T}, \mathbf{X}_{t-1}) + \frac{\partial g(\boldsymbol{\theta}_t, \mathbf{X}_{t-1})}{\partial \boldsymbol{\theta}_t} \Big|_{(\boldsymbol{\theta}_t = \boldsymbol{\theta}_{t|T})} (\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t|T}) + \dots \quad (10)$$

where the smoothing estimate $\boldsymbol{\theta}_{t|T}$ denotes the conditional mean of $\boldsymbol{\theta}_t$ given $y_{1:T} = \{y_1, \dots, y_T\}$.

The EKF utilizes only the first-order Taylor expansion, so we can obtain

$$g(\boldsymbol{\theta}_t, \mathbf{X}_{t-1}) \approx g(\boldsymbol{\theta}_{t|T}, \mathbf{X}_{t-1}) + \left. \frac{\partial g(\boldsymbol{\theta}_t, \mathbf{X}_{t-1})}{\partial \boldsymbol{\theta}_t} \right|_{(\boldsymbol{\theta}_t = \boldsymbol{\theta}_{t|T})} (\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t|T}). \quad (11)$$

Under the basic assumptions of the EKF (no model and parameter uncertainty, zero-mean white-noise sequence, known process and measurement models, etc.), the smoothing estimates are unbiased, and the smoothing estimates satisfy

$$E(\boldsymbol{\theta}_{t|T}) = \boldsymbol{\theta}_t. \quad (12)$$

So we can get

$$E(g(\boldsymbol{\theta}_t, \mathbf{X}_{t-1})) \approx g(\boldsymbol{\theta}_{t|T}, \mathbf{X}_{t-1}). \quad (13)$$

Subsequently, we compute the covariance of $g(\boldsymbol{\theta}_t, \mathbf{X}_{t-1})$

$$\begin{aligned} E \left[(g(\boldsymbol{\theta}_t, \mathbf{X}_{t-1}) - g(\boldsymbol{\theta}_{t|T}, \mathbf{X}_{t-1})) \right. \\ \left. (g(\boldsymbol{\theta}_t, \mathbf{X}_{t-1}) - g(\boldsymbol{\theta}_{t|T}, \mathbf{X}_{t-1}))^T \right] \\ \approx E \left[\left[\left. \frac{\partial g(\boldsymbol{\theta}_t, \mathbf{X}_{t-1})}{\partial \boldsymbol{\theta}_t} \right|_{(\boldsymbol{\theta}_t = \boldsymbol{\theta}_{t|T})} (\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t|T}) \right] \right. \\ \left. \times \left[\left. \frac{\partial g(\boldsymbol{\theta}_t, \mathbf{X}_{t-1})}{\partial \boldsymbol{\theta}_t} \right|_{(\boldsymbol{\theta}_t = \boldsymbol{\theta}_{t|T})} (\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t|T}) \right]^T \right] \\ = \mathbf{G}_t \mathbf{P}_{t|T} \mathbf{G}_t^T. \end{aligned} \quad (14)$$

Hence, it follows that:

$$\begin{aligned} E \left[g(\boldsymbol{\theta}_t, \mathbf{X}_{t-1}) (g(\boldsymbol{\theta}_t, \mathbf{X}_{t-1}))^T \right] \\ \approx \mathbf{G}_t \mathbf{P}_{t|T} \mathbf{G}_t^T + g(\boldsymbol{\theta}_{t|T}, \mathbf{X}_{t-1}) (g(\boldsymbol{\theta}_{t|T}, \mathbf{X}_{t-1}))^T \end{aligned} \quad (15)$$

where \mathbf{G}_t corresponds to the Jacobian matrix of the measurement function:

$$\begin{aligned} \mathbf{G}_t &= \left. \frac{\partial g(\boldsymbol{\theta}_t, \mathbf{X}_{t-1})}{\partial \boldsymbol{\theta}_t} \right|_{(\boldsymbol{\theta}_t = \boldsymbol{\theta}_{t|T})} \\ &= \left[\frac{\partial g(\boldsymbol{\theta}_t, \mathbf{X}_{t-1})}{\partial \theta_{1,t}} \quad \frac{\partial g(\boldsymbol{\theta}_t, \mathbf{X}_{t-1})}{\partial \theta_{2,t}} \quad \dots \quad \frac{\partial g(\boldsymbol{\theta}_t, \mathbf{X}_{t-1})}{\partial \theta_{l,t}} \right] \end{aligned} \quad (16)$$

and $\mathbf{P}_{t|T}$ denotes the conditional covariance of $\boldsymbol{\theta}_t$ given $y_{1:T} = \{y_1, \dots, y_T\}$:

$$\mathbf{P}_{t|T} = E \left[(\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t|T}) (\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t|T})^T \right]. \quad (17)$$

Equations (13)–(15) are substituted in (9), and then we get

$$\begin{aligned} E[\ln p(\boldsymbol{\theta}_{0:T}, y_{1:T} | \boldsymbol{\varphi})] &= -\frac{1}{2} \ln |\boldsymbol{\Xi}_0| - \frac{T}{2} \ln |\mathbf{Q}| \\ &\quad - \frac{T}{2} \ln |R| - \frac{1}{2} \text{tr} \left\{ \boldsymbol{\Xi}_0^{-1} [\boldsymbol{\theta}_{0|T} \boldsymbol{\theta}_{0|T}^T \right. \\ &\quad \left. - 2\boldsymbol{\theta}_{0|T} \boldsymbol{\mu}_0^T + \boldsymbol{\mu}_0 \boldsymbol{\mu}_0^T + \mathbf{P}_{0|T}] \right\} \end{aligned}$$

$$\begin{aligned} &- \frac{1}{2} \sum_{t=1}^T \text{tr} \left\{ \mathbf{Q}^{-1} [\boldsymbol{\theta}_{t|T} \boldsymbol{\theta}_{t|T}^T + \mathbf{P}_{t|T} \right. \\ &\quad \left. - 2(\boldsymbol{\theta}_{t|T} \boldsymbol{\theta}_{t-1|T}^T + \mathbf{P}_{t,t-1|T})^T \right. \\ &\quad \left. + \boldsymbol{\theta}_{t-1|T} \boldsymbol{\theta}_{t-1|T}^T + \mathbf{P}_{t-1|T}] \right\} \\ &\quad - \frac{(l+n)T+l}{2} \ln 2\pi \\ &\quad - \frac{1}{2} \sum_{t=1}^T \text{tr} \left\{ R^{-1} [(y_t - g(\boldsymbol{\theta}_{t|T}, \mathbf{X}_{t-1})) \right. \\ &\quad \left. \times (y_t - g(\boldsymbol{\theta}_{t|T}, \mathbf{X}_{t-1}))^T + \mathbf{G}_t \mathbf{P}_{t|T} \mathbf{G}_t^T] \right\}. \end{aligned} \quad (18)$$

Smoothing often includes the forward and backward filtering over a segment of data so as to obtain improved estimates. The forward filtering stage involves computing the estimates $\boldsymbol{\theta}_{t|t}$ and $\mathbf{P}_{t|t}$ over a segment of T samples, where $\boldsymbol{\theta}_{t|t}$ and $\mathbf{P}_{t|t}$ denote the conditional mean and conditional covariance of $\boldsymbol{\theta}_t$ given $y_{1:t} = \{y_1, \dots, y_t\}$. Then, the EKF-based forward filtering for estimating model (4) can be derived as follows. The predicted state vector:

$$\boldsymbol{\theta}_{t|t-1} = E[\boldsymbol{\theta}_t | y_{1:t-1}] = \boldsymbol{\theta}_{t-1|t-1}. \quad (19)$$

The predicted conditional covariance of $\boldsymbol{\theta}_t$:

$$\begin{aligned} \mathbf{P}_{t|t-1} &= E \left[(\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t|t-1}) (\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t|t-1})^T \right] \\ &= \mathbf{P}_{t-1|t-1} + \mathbf{Q}. \end{aligned} \quad (20)$$

The Kalman gain:

$$\mathbf{K}_t = \mathbf{P}_{t|t-1} \mathbf{G}_t^T (\mathbf{G}_t \mathbf{P}_{t|t-1} \mathbf{G}_t^T + R)^{-1}. \quad (21)$$

Updated system state estimate:

$$\begin{aligned} \boldsymbol{\theta}_{t|t} &= E[\boldsymbol{\theta}_t | y_{1:t}] \\ &= \boldsymbol{\theta}_{t|t-1} + \mathbf{K}_t [y_t - g(\boldsymbol{\theta}_{t|t-1}, \mathbf{X}_{t-1})]. \end{aligned} \quad (22)$$

Updated estimate covariance:

$$\begin{aligned} \mathbf{P}_{t|t} &= E \left[(\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t|t}) (\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t|t})^T \right] \\ &= \mathbf{P}_{t|t-1} - \mathbf{K}_t \mathbf{G}_t \mathbf{P}_{t|t-1}. \end{aligned} \quad (23)$$

To obtain the smoothing estimates $\boldsymbol{\theta}_{t|T}$ and $\mathbf{P}_{t|T}$, we employ the Rauch-Tung-Strieber smoother to do the following backward recursions

$$\begin{cases} \mathbf{J}_t = \mathbf{P}_{t|t} \mathbf{P}_{t+1|t}^{-1} \\ \boldsymbol{\theta}_{t|T} = \boldsymbol{\theta}_{t|t} + \mathbf{J}_t (\boldsymbol{\theta}_{t+1|T} - \boldsymbol{\theta}_{t+1|t}) \\ \mathbf{P}_{t|T} = \mathbf{P}_{t|t} + \mathbf{J}_t (\mathbf{P}_{t+1|T} - \mathbf{P}_{t+1|t}) \mathbf{J}_t^T. \end{cases} \quad (24)$$

We also require the cross-covariance $\mathbf{P}_{t,t-1|T}$, which is defined as follows

$$\mathbf{P}_{t,t-1|T} = E \left[(\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t|T}) (\boldsymbol{\theta}_{t-1} - \boldsymbol{\theta}_{t-1|T})^T \right]. \quad (25)$$

And it can be obtained through the backward recursions

$$\mathbf{P}_{t,t-1|T} = \mathbf{P}_{t|t} \mathbf{J}_{t-1}^T + \mathbf{J}_t (\mathbf{P}_{t+1,t|T} - \mathbf{P}_{t|t}) \mathbf{J}_{t-1}^T. \quad (26)$$

The backward recursions as above are initialized as following

$$\begin{cases} \boldsymbol{\theta}_{T|T} = \boldsymbol{\theta}_T \\ \mathbf{P}_{T|T} = \mathbf{P}_T \\ \mathbf{P}_{T,T-1|T} = (\mathbf{I} - \mathbf{K}_T \mathbf{G}_T) \mathbf{P}_{T-1|T-1}. \end{cases} \quad (27)$$

Compared with the filtering algorithm that uses the observations up to time t for estimation of the state θ_t , the smoothing yields a more accurate estimate $\theta_{t|T}$ by using all available data up to time T .

To find the optimal parameters φ , we need to maximize the expected value of the log-likelihood with respect to the parameters, and then compute the derivatives with respect to each parameter individually. That is

$$\begin{cases} \frac{\partial}{\partial \mu_0} E[\ln p(\theta_{0:T}, y_{1:T} | \varphi)] = 0 \\ -\frac{1}{2} \frac{\partial}{\partial \mu_0} \text{tr} \{ \Xi_0^{-1} [(\theta_{0|T} - \mu_0)(\theta_{0|T} - \mu_0)^T + \mathbf{P}_{0|T}] \} = 0 \\ \Xi_0^{-1} (\theta_{0|T} - \mu_0) = 0 \\ \mu_0 = \theta_{0|T} \end{cases} \quad (28)$$

$$\begin{cases} \frac{\partial}{\partial \Xi_0^{-1}} E[\ln p(\theta_{0:T}, y_{1:T} | \varphi)] = 0 \\ \frac{1}{2} \frac{\partial}{\partial \Xi_0^{-1}} \left\{ \ln |\Xi_0^{-1}| - \text{tr} [\Xi_0^{-1} ((\theta_{0|T} - \mu_0)(\theta_{0|T} - \mu_0)^T + \mathbf{P}_{0|T})] \right\} = 0 \\ \frac{\Xi_0}{2} - \frac{1}{2} ((\theta_{0|T} - \mu_0)(\theta_{0|T} - \mu_0)^T + \mathbf{P}_{0|T}) = 0 \\ \Xi_0 = \mathbf{P}_{0|T} \end{cases} \quad (29)$$

$$\begin{cases} \frac{\partial}{\partial \mathbf{Q}^{-1}} E[\ln p(\theta_{0:T}, y_{1:T} | \varphi)] = 0 \\ \frac{\partial}{\partial \mathbf{Q}^{-1}} \left\{ \frac{T}{2} \ln |\mathbf{Q}^{-1}| - \frac{1}{2} [\text{tr}(\mathbf{Q}^{-1} (\mathbf{\Gamma} - 2\mathbf{\Upsilon}^T + \mathbf{\Lambda}))] \right\} = 0 \\ \frac{T}{2} \mathbf{Q} - \frac{1}{2} (\mathbf{\Gamma} - 2\mathbf{\Upsilon}^T + \mathbf{\Lambda}) = 0 \\ \mathbf{Q} = \frac{1}{T} (\mathbf{\Gamma} - 2\mathbf{\Upsilon}^T + \mathbf{\Lambda}) \end{cases} \quad (30)$$

$$\begin{cases} \frac{\partial}{\partial R^{-1}} E[\ln p(\theta_{0:T}, y_{1:T} | \varphi)] = 0 \\ \frac{\partial}{\partial R^{-1}} \left\{ \frac{T}{2} \ln |R^{-1}| - \frac{1}{2} \sum_{t=1}^T \text{tr} [R^{-1} ((y_t - g(\theta_{t|T}, \mathbf{X}_{t-1})) \times (y_t - g(\theta_{t|T}, \mathbf{X}_{t-1}))^T + \mathbf{G}_t \mathbf{P}_{t|T} \mathbf{G}_t^T)] \right\} = 0 \\ \frac{T}{2} R - \frac{1}{2} \sum_{t=1}^T \left[(y_t - g(\theta_{t|T}, \mathbf{X}_{t-1})) (y_t - g(\theta_{t|T}, \mathbf{X}_{t-1}))^T + \mathbf{G}_t \mathbf{P}_{t|T} \mathbf{G}_t^T \right] = 0 \\ R = \frac{1}{T} \sum_{t=1}^T \left[(y_t - g(\theta_{t|T}, \mathbf{X}_{t-1})) (y_t - g(\theta_{t|T}, \mathbf{X}_{t-1}))^T + \mathbf{G}_t \mathbf{P}_{t|T} \mathbf{G}_t^T \right] \end{cases} \quad (31)$$

where

$$\begin{cases} \mathbf{\Gamma} = \sum_{t=1}^T (\theta_{t|T} \theta_{t|T}^T + \mathbf{P}_{t|T}) \\ \mathbf{\Lambda} = \sum_{t=1}^T (\theta_{t-1|T} \theta_{t-1|T}^T + \mathbf{P}_{t-1|T}) \\ \mathbf{\Upsilon} = \sum_{t=1}^T (\theta_{t|T} \theta_{t-1|T}^T + \mathbf{P}_{t,t-1|T}). \end{cases} \quad (32)$$

It is significant to mention that the EM algorithm is applied to obtain maximum likelihood (ML) estimates of the above parameters and states, which can reduce the computational complexity and guarantee the convergence to a stationary point while continuously increasing the ML function. But EM algorithm is computationally expensive when the state dimension is high.

4 Simulation Results

To evaluate the performance of the presented approach, we predict the well-known Mackey-Glass time series through three cases. In the first case, the data is generated from the following Mackey-Glass equation:

$$\frac{dy_t}{dt} = \frac{ay_{t-\tau_0}}{1 + y_{t-\tau_0}^c} - by_t \quad (33)$$

where the parameters are chosen to be $a = 0.2$, $b = 0.1$, $c = 10$ and $\tau_0 = 20$. A thousand values are sampled and the series is shown in Fig. 2. This chaotic benchmark time series was studied in [1], [2]. We use the RBF-AR(p, m, d) model to predict the nonlinear time series, where the parameters p, m and d are defined as shown in model (2). The first 500 data points are used to train the model, and the last 500 data are used to test the model.

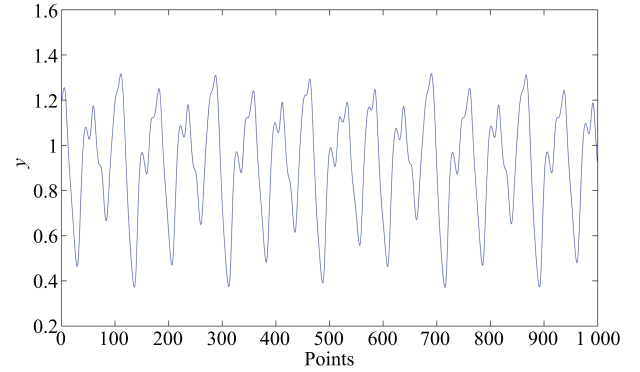


Fig. 2. The original Mackey-Glass time series.

To make the comparisons between the SNPOM, the EKF and the EM-EKF, we predict the value y_t from the fixed input vector $[y_{t-5}, y_{t-4}, y_{t-3}, y_{t-2}, y_{t-1}]$. Thus, an RBF-AR(5,3,2) model is used to predict the one-step ahead output of this complex nonlinear time series as follows:

$$\begin{cases} y_t = \phi_0(\mathbf{X}_{t-1}) + \sum_{i=1}^5 \phi_i(\mathbf{X}_{t-1}) y_{t-i} + e_t \\ \phi_i(\mathbf{X}_{t-1}) = \omega_{i,0} + \sum_{k=1}^3 \omega_{i,k} \exp \{ -\lambda_k \|\mathbf{X}_{t-1} - \mathbf{Z}_k\|_2^2 \} \\ \mathbf{X}_{t-1} = [y_{t-1}, y_{t-2}]^T. \end{cases} \quad (34)$$

Table I reports the comparison results of the proposed EM-EKF, the EKF and the SNPOM. The estimated results are the mean squared errors (MSEs), observation noise variance and the standard deviations (given in the parentheses). It is worth mentioning that Table I gives two different estimation results using EKF in the cases of given different measurement noise variances. From Table I we can see that the MSEs of the training data using EM-EKF are lesser than those using SNPOM and EKF, while the testing data is only slightly lesser. We attribute this to the

fact that the SNPOM has good estimation results for clean data. Besides, the SNPOM obtains slightly better results than does the EKF in these conditions of given different measurement noise variances ($R = 0.002$ or $R = 0.0002$). Furthermore, by comparison between the two conditions in the EKF, the closer the given measurement noise variance is to the true value ($R = 0$), the better the results are. This illustrates that setting the measurement noise appropriately leads to success or failure when using EKF. The estimate of observation noise variance using EM-EKF is close to the true value (zero), while the estimate of that using SNPOM is unknown, and the observation noise variance in the EKF must be given in advance. In Fig. 3, the top plot shows the increase of the log-likelihood at each step, the middle plot shows the convergence of the measurements noise variance R , and the bottom plot shows the trace of the process noise variance Q . The initial conditions for the state θ_0 and the measurements noise variance R are randomly chosen with a uniform distribution from the range $[0,1]$ and $[0, 0.01]$, respectively, and matrices Q and P are initialized as identity matrix and 100 times identity matrix, respectively.

TABLE I
COMPARISON RESULTS FOR MACKEY-GLASS TIME SERIES

Method	MSE (Training)	MSE (Testing)	R
SNPOM	1.0800E-7	1.2600E-7	unknown
EKF	1.9560E-7	2.1786E-7	0.002 (given)
EKF	1.2559E-7	1.9793E-7	0.0002 (given)
EM-EKF	7.1765E-8	1.2008E-7	1.8146E-7

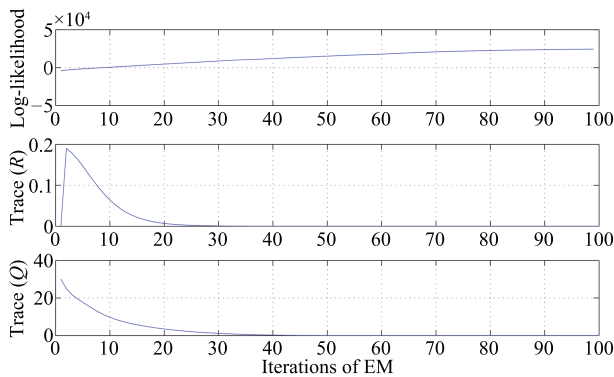


Fig. 3. The convergence process of the Log-likelihood, the measurement and process noise variance.

To examine the noise effect in the time series, the EM-EKF, the EKF and the SNPOM are used to estimate the Mackey-Glass chaotic time series corrupted by additive white noise with variance 0.25 and 1 in the following two cases (see Fig. 4). Table II gives the performance of the EM-EKF compared to those of the SNPOM and the EKF on estimating the Mackey-Glass chaotic time series corrupted by additive white noise. From Table II, one can see that the overall performance obtained by the EM-EKF is better than that of the SNPOM and the EKF. In detail, the EM-EKF has smaller values of MSEs than those of the SNPOM and the EKF for the training data and testing data, although the improvement is not significant. As for the EKF, when the given values of the measurement noise variance ($R = 0.1$ and $R = 0.6$) deviate relatively far from the true values ($R = 0.25$ and $R = 1$), the results become

worse, even worse than those obtained by the SNPOM. On the contrary, when the given values ($R = 0.2$ and $R = 0.8$) are relatively close to the true values, the EKF obtains slightly better results than does the SNPOM, but still obtains worse results than does the EM-EKF. This illustrates that setting of the measurement noise plays an important part in the use of the EKF again. In both cases, the values of measurements noise variance R using EM-EKF are 0.25950 and 1.0589, respectively, which are very close to the true values. In contrast, the SNPOM can not estimate the values of measurement noise variance, and the values of measurements noise variance should be given beforehand in the use of the EKF. The initial conditions are the same as case one except that R is randomly chosen with a uniform distribution from the range $[0,1]$. Figs. 5 and 6 show the convergent process of the log-likelihood and noise variance.

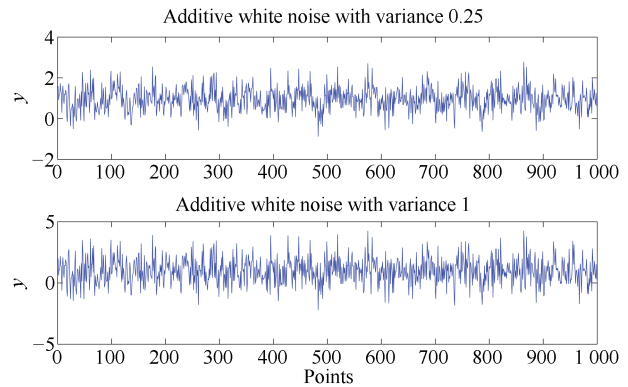


Fig. 4. The Mackey-Glass chaotic time series corrupted by additive white noise.

TABLE II
COMPARISON RESULTS FOR MACKEY-GLASS TIME SERIES
CORRUPTED BY ADDITIVE WHITE NOISE

Noise variance	Method	MSE (Training)	MSE (Testing)	R
0.25	SNPOM	0.26606	0.28120	unknown
	EKF	0.26577	0.28082	0.2 (given)
	EKF	0.27343	0.28825	0.1 (given)
1	EM-EKF	0.25750	0.27825	0.25950
	SNPOM	0.97452	1.1644	unknown
	EKF	0.97215	1.1512	0.8 (given)
	EKF	0.98262	1.1785	0.6 (given)
	EM-EKF	0.96590	1.13907	1.0589

Statistical F test is used to judge whether the error variance estimated by the EM-EKF is equal to the variance estimated by the SNPOM. As shown in Table III, the values of the computing F test statistic (F) are greater than the table value ($F_{\alpha=0.05}$) except for the testing data without additional white noise, so the null hypothesis that two variances are equal is rejected at level $\alpha = 0.05$. This indicates that there are significant differences between the above two methods and the performance of the EM-EKF is superior to the SNPOM.

To compare the computational complexity, Table IV lists the computational time for the three methods. The simulations are implemented on a computer (Inter (R) Core (TM)2 Duo CPU E7200 @2.53 GHz, 8 G-RAM). The av-

erage running time (100 iterations) for EM-EKF is approximately 30.766 seconds, for SNPOM (100 iterations) 10.690 seconds, and for EKF 0.40572 seconds. Obviously, the EM-EKF is more time-consuming than the other two methods since EM algorithm is computationally expensive for high-dimension states. Also, for a large RBF network, the computational expense of the EKF could be burdensome and it increases linearly with the number of training samples. Instead, the SNPOM is a hybrid method, depending partly on the Levenberg-Marquardt method (LMM) for nonlinear parameter optimization and partly on the least-squares method (LSM) for linear parameter estimation. The SNPOM can greatly accelerate the computational convergence of the parameter search process, especially for the RBF-type models with larger number of linear weights and smaller number of nonlinear parameters. However, the EM-EKF, as an alternative way, is superior to the SNPOM and the EKF and can accurately estimate the noise variance.

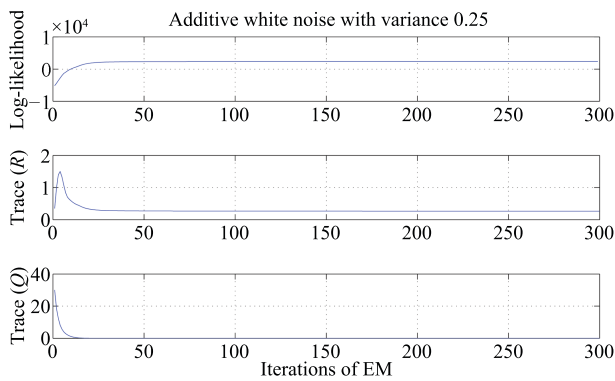


Fig. 5. The convergence process of the Log-likelihood, the measurement and process noise variance.

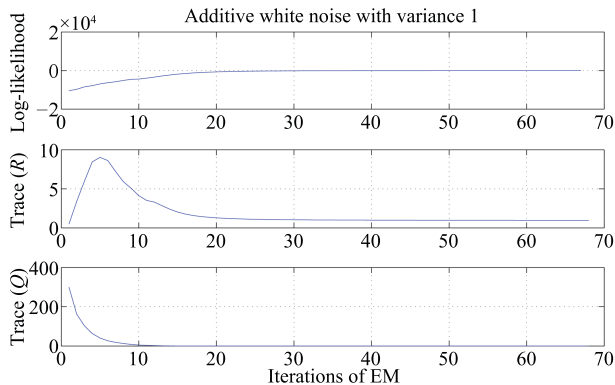


Fig. 6. The convergence process of the Log-likelihood, the measurement and process noise variance.

TABLE III
THE RESULTS OF STATISTICAL F TEST AT LEVEL 0.05

Case	$F_{\alpha=0.05}$	F	Results
Case 1 (Training)	1.16	1.4642	Reject; Difference
Case 1 (Testing)	1.16	1.0105	No reject; No difference
Case 2 (Training)	1.16	14.1212	Reject; Significant difference
Case 2 (Testing)	1.16	9.2984	Reject; Significant difference
Case 3 (Training)	1.16	37.8297	Reject; Significant difference
Case 3 (Testing)	1.16	15.5751	Reject; Significant difference

TABLE IV
THE COMPUTATION TIME OF DIFFERENT METHODS (s)

Method	Time
SNPOM	10.690
EKF	0.40572
EM-EKF	30.766

As a whole, the EM-EKF method is capable of estimating the parameters of the RBF-AR model, and the initial conditions and the noise variances are identified by use of the EM algorithm, which can further improve the modeling precision. Comparison results indicate that the RBF-AR model estimated by the EM-EKF makes more accurate predictions than do the SNPOM and the EKF, although the values of the MSEs using EM-EKF are only slightly smaller than those of the SNPOM and EKF in some cases. However, F test shows there is significant difference between results obtained by the SNPOM and the EM-EKF. Furthermore, the estimation of observation noise variance using EM-EKF is close to the true value, while the estimate of that using SNPOM is unknown, and the observation noise variance in the EKF must be given in advance. Therefore, we can conclude that the EM-EKF method is an advisable choice for estimating RBF-AR model and is especially appropriate for signals disturbed by noise.

5 Conclusion

In this paper, the EM-EKF method is developed to estimate the parameter of RBF-AR model. Firstly, the model is reconstructed as a new type of general radial basis function neural networks. Secondly, to circumvent the EKF's limitation of unknown prior knowledge, the EM is proposed to calculate the initial states and the measurement and process noise variance. By combining the EM and extended Kalman filtering and smoothing process, the EM-EKF method is proposed to estimate the parameters of the RBF-AR model, the initial conditions and the noise variances jointly, and can further improve the modeling precision. Comparisons of the performance of the EM-EKF with the SNPOM and the EKF are performed, and the results indicate that the RBF-AR model estimated by the EM-EKF makes more accurate predictions than do the SNPOM and the EKF, although the EM-EKF is more time-consuming. Moreover, the estimate of observation variance converges to the true value. Finally, F test indicates there is significant difference between results obtained by the SNPOM and the EM-EKF. Our future work would develop the EM-EKF algorithm for RBF-ARX (Peng *et al.* [2]) estimation and apply the RBF-AR model based on the EM-EKF algorithm to other types of time series and complex systems analysis.

References

- 1 Z. Y. Shi, Y. Tamura, and T. Ozaki, "Nonlinear time series modelling with the radial basis function-based state-dependent autoregressive model," *Int. J. Syst. Sci.*, vol. 30, no. 7, pp. 717–727, Jul. 1999.
- 2 H. Peng, T. Ozaki, V. Haggan-Ozaki, and Y. Toyoda, "A parameter optimization method for radial basis function type models," *IEEE Trans. Neural Netw.*, vol. 14, no. 2, pp. 432–438, Mar. 2003.
- 3 M. Gan, H. Peng, X. Y. Peng, X. H. Chen, and G. Inoussa, "A locally linear RBF network-based state-dependent AR

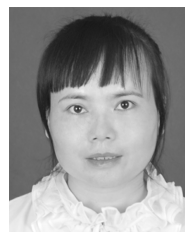
- model for nonlinear time series modeling," *Inform. Sci.*, vol. 180, no. 22, pp. 4370–4380, Nov. 2010.
- 4 M. Gan, C. L. P. Chen, H. X. Li, and L. Chen, "Gradient radial basis function based varying-coefficient Autoregressive model for nonlinear and nonstationary time series," *IEEE Signal Proc. Lett.*, vol. 22, no. 7, pp. 809–812, Jul. 2015.
 - 5 H. Peng, J. Wu, G. Inoussa, Q. L. Deng, and K. Nakano, "Nonlinear system modeling and predictive control using the RBF nets-based quasi-linear ARX model," *Control Eng. Pract.*, vol. 17, no. 1, pp. 59–66, Jan. 2009.
 - 6 H. Peng, G. Kitagawa, J. Wu, and K. Ohtsu, "Multivariable RBF-ARX model-based robust MPC approach and application to thermal power plant," *Appl. Math. Model.*, vol. 35, no. 7, pp. 3541–3551, Jul. 2011.
 - 7 M. Gan, H. Peng, and L. Y. Chen, "A global-local optimization approach to parameter estimation of RBF-type models," *Inform. Sci.*, vol. 197, pp. 144–160, Aug. 2012.
 - 8 M. Gan, H. X. Li, and H. Peng, "A variable projection approach for efficient estimation of RBF-ARX model," *IEEE Trans. Cybern.*, vol. 45, no. 3, pp. 462–471, Mar. 2015.
 - 9 H. K. Wei and S. I. Amari, "Dynamics of learning near singularities in radial basis function networks," *Neural Netw.*, vol. 21, no. 7, pp. 989–1005, Sep. 2008.
 - 10 J. S. Kim and S. Jung, "Implementation of the RBF neural chip with the back-propagation algorithm for on-line learning," *Appl. Soft Comput.*, vol. 29, pp. 233–244, Apr. 2015.
 - 11 F. P. Härter and H. F. de Campos Velho, "New approach to applying neural network in nonlinear dynamic model," *Appl. Math. Model.*, vol. 32, no. 12, pp. 2621–2633, Dec. 2008.
 - 12 H. Z. Yang, J. Li, and F. Ding, "A neural network learning algorithm of chemical process modeling based on the extended Kalman filter," *Neurocomputing*, vol. 70, no. 4–6, pp. 625–632, Jan. 2007.
 - 13 K. Salahshoor, S. Zakeri, and M. H. Sefat, "Stabilization of gas-lift oil wells by a nonlinear model predictive control scheme based on adaptive neural network models," *Eng. Appl. Artif. Intel.*, vol. 26, no. 8, pp. 1902–1910, Sep. 2013.
 - 14 N. Y. Zeng, Z. D. Wang, Y. R. Li, M. Du, and X. H. Liu, "Inference of nonlinear state-space models for sandwich-type lateral flow immunoassay using extended Kalman filtering," *IEEE Trans. Biomed. Eng.*, vol. 58, no. 7, pp. 1959–1966, Jul. 2011.
 - 15 J. F. G. de Freitas, M. Niranjana, and A. H. Gee, "Hierarchical Bayesian models for regularization in sequential learning," *Neural Comput.*, vol. 12, no. 4, pp. 933–953, Apr. 2000.
 - 16 Y. S. T. Hong, "Dynamic nonlinear state-space model with a neural network via improved sequential learning algorithm for an online real-time hydrological modeling," *J. Hydrol.*, vol. 468–469, pp. 11–21, Oct. 2012.
 - 17 Y. H. Xi, H. Peng, and X. H. Chen, "A sequential learning algorithm based on adaptive particle filtering for RBF networks," *Neural Comput. Appl.*, vol. 25, no. 3–4, pp. 807–814, Sep. 2014.
 - 18 N. Y. Zeng, Z. D. Wang, Y. R. Li, M. Du, and X. H. Liu, "A hybrid EKF and switching PSO algorithm for joint state and parameter estimation of lateral flow immunoassay models," *IEEE/ACM Trans. Comput. Biol. Bioinform.*, vol. 9, no. 2, pp. 321–329, Mar.–Apr. 2012.
 - 19 N. Y. Zeng, Z. D. Wang, Y. R. Li, M. Du, and X. H. Liu, "Identification of nonlinear lateral flow immunoassay state-space models via particle filter approach," *IEEE Trans. Nanotechnol.*, vol. 11, no. 2, pp. 321–327, Mar. 2012.
 - 20 M. Hürzeler and H. R. Künsch, "Approximating and maximising the likelihood for a general state-space model," in *Sequential Monte Carlo Methods in Practice*, A. Doucet, N. de Freitas, and N. Gordon, Eds. New York: Springer, 2001, pp. 159–175.
 - 21 N. Y. Zeng, Z. D. Wang, Y. R. Li, M. Du, J. Cao, and X. H. Liu, "Time series modeling of nano-gold immunochromatographic assay via expectation maximization algorithm," *IEEE Trans. Biomed. Eng.*, vol. 60, no. 12, pp. 3418–3424, Dec. 2013.
 - 22 M. Lázaro, I. Santamaría, and C. Pantaleón, "A new EM-based training algorithm for RBF networks," *Neural Netw.*, vol. 16, no. 1, pp. 69–77, Jan. 2003.
 - 23 C. Constantinopoulos and A. Likas, "Semi-supervised and active learning with the probabilistic RBF classifier," *Neurocomputing*, vol. 71, no. 13–15, pp. 2489–2498, Aug. 2008.



Yanhui Xi received the Ph.D. degree in control science and engineering from Central South University, China, in 2013. She is an Associate Professor at Changsha University of Science and Technology. Her research interests include system modeling and identification, nonlinear optimization, and traveling identification. Corresponding author of this paper. E-mail: xiyanhui@126.com



Hui Peng received the Ph.D. degree at the School of Mathematical and Physical Science, the Graduate University for Advanced Studies, Japan, in 2003. He is a Professor at the School of Information Science and Engineering, Central South University. His research interests include predictive control, adaptive control, system modeling and identification, modeling and dynamic asset allocation in financial markets. E-mail: huipeng@mail.csu.edu.cn



Hong Mo received the Ph.D. degree from Graduate University of Chinese Academy of Sciences in 2004. She is a Professor at the School of Electrical and Information Engineering, Changsha University of Sciences and Technology. Her research interests include linguistic dynamic systems, time-varying universe, and type-2 fuzzy sets. E-mail: mohong198@163.com