CIRCUITS SYSTEMS SIGNAL PROCESSING
VOL. 20, NO. 5, 2001, PP. 575–597

STATE-DEPENDENT VECTOR HYBRID LINEAR AND NONLINEAR ARMA MODELING: APPLICATIONS*

Yuanjin Zheng,[1] *Zhiping Lin*[1] *and David B. H. Tay*[2]

Abstract. In a recent companion paper, a new method has been presented for modeling general vector nonstationary and nonlinear processes based on a state-dependent vector hybrid linear and nonlinear autoregressive moving average (SVH-ARMA) model. This paper discusses some potential applications of the SVH-ARMA model, including signal filtering, time series prediction, and system control. First, a state-space model governed by a hidden Markov Chain is shown to be equivalent to the SVH-ARMA model. Based on this state-space model, the extended Kalman filtering and Bayesian estimation techniques are applied for noisy signal enhancement. The result of a noisy image enhancement verifies that the model can track the time-varying statistical characteristics of nonstationary and nonlinear processes adaptively. Second, the SVH-ARMA model is used for a vector time series prediction, which can attain more accurate multiple step ahead prediction, than conventional forecasting methods. Third, a new technique is developed for predicting scalar long correlation time series in the wavelet scale space domain based on the SVH-ARMA model. Dyadic wavelet transform is employed to convert a scalar time series to a vector time series, to which the SVH-ARMA model is applied for vector time series prediction. More accurate and robust forecasting results in both one step and multiple step ahead prediction can be gained. See also the companion paper on theory, by Zheng et al., pp. 551–574, this issue.

Key words: SVH-ARMA model, extended Kalman filtering, long correlation time series, one step ahead prediction, multiple step ahead prediction, dyadic wavelet transform.

1. Introduction

A general signal process may not be accurately modeled by only a linear model because of the potential nonlinear correlation between successive signal samples.

* Received January 10, 2000; revised September 7, 2000.

[1] School of Electrical and Electronic Engineering, S2, Nanyang Technological University, Nanyang Avenue, Singapore, 639798. E-mail for Zheng: yuanjin@ime.org.sg, E-mail for Lin: EZPLin@ntu.edu.sg

[2] Department of Electronic Engineering, Latrobe University, Bundoora, Victoria 3083, Australia. E-mail: D.Tay@ee.latrobe.edu.au

Many signal processes such as speech, medical images, and economic time series display this feature [5], [17], [19]. The merit of the state-dependent vector hybrid linear and nonlinear ARMA (SVH-ARMA) model presented in a recent companion paper [22] is that it can automatically and adaptively represent the linear and/or nonlinear parts of a nonstationary process. Further, this model can be converted to a Markov chain switched state-space model. By this state-space model, the possible nonstationary varying of the process can be tracked by the Bayesian posterior estimation of the distribution of Markov chain state evolution. Moreover, the equivalent state-space model possesses both linear and nonlinear (approximated by Taylor series expansion) components and can be used to optimally filter and smooth the linear and/or nonlinear components of a noisy signal process separately and adaptively [1].

Neither a linear nor a nonlinear ARMA model alone can track the long correlation features of time series processes [7]. The long correlation of a random process means that the covariance function of the process decays to zero very slowly. Therefore, the value of a sample may be affected substantially by samples far away from this sample. Many time series such as sun spots, speech, stocks, exchange rates, and other chaotic time series show strong long-range temporal correlation structures [5]. It has been reported that a model combining a linear AR model with a compressive nonlinear model would possess a certain ability to represent the long second-order correlation characteristics of time series [7]. This discovery is extended in this paper. A vector linear ARMA model is integrated with a vector nonlinear ARMA model to construct a hybrid ARMA model. Although a linear model or a nonlinear model alone cannot produce long correlation characteristics, the interaction between the linear model and the nonlinear model makes it possible for the hybrid model to track the long-range dependence of vector time series. Moreover, the time series in reality are not always stationary. For example, some time series in economics such as stock returns, and exchange rates are stationary within a regime, but globally nonstationary. To track this nonstationary varying of the processes, a state-dependent model may be employed. The model parameters change adaptively with the varying of statistical characteristics of the process. In this paper, by virtue of the prediction of the Markov chain hidden in the SVH-ARMA model [10], it is possible to attain dynamic self-tuning short-range or long-range predictions.

From the point of view of time series prediction, many methods can only attain accurate one step ahead prediction. For instance, the famous Box–Jenkins strategy can achieve the optimal linear vector one step ahead prediction [4]. The sample delay multilayer feedforward (MLFF) neural networks or recurrent neural networks (RNN) techniques can attain the optimal nonlinear vector one step ahead prediction [15]. Unfortunately, neither of these approaches possesses good long-range and multiple step ahead prediction ability because of their exponentially decreasing second-order correlation characteristics. It has been verified by extensive simulations that our SVH-ARMA model can track the tendency of long correlation time series and thus may attain more accurate and robust multiple step ahead prediction.

Wavelets have been combined with neural networks for function learning and time series modeling [8], [20]. We can decompose a one-dimensional scalar signal into signals with different scales by dyadic wavelet transform [13] and then group these signals along scales to form a vector signal. This vector signal represents the multiresolution features of the original scalar signal and may be modeled by the SVH-ARMA model. In this way, the problem of scalar time series prediction can be solved through vector time series prediction in the wavelet transform domain. Simulation results verify that this method for scalar time series prediction greatly improves the accuracy of one step ahead prediction and makes multiple step ahead prediction more robust.

The rest of the paper is organized as follows. The various potential applications of the SVH-ARMA model are summarized in Section 2. In Section 3, extended Kalman filtering based on the SVH-ARMA model for signal enhancement is derived. In Section 4, the SVH-ARMA model for long correlation scalar and vector time series prediction is discussed. The simulation results are shown in Section 5.

2. SVH-ARMA (ARMAX) model for filtering, prediction, and control

Some potential applications of the SVH-ARMA model and the SVH-ARMAX (autoregressive moving average with exogenous input) model are summarized in this section.

2.1. *Filtering*

If the original signal process $\{\mathbf{y}_t\}_{t=1,\dots,T}$ is contaminated by a white noise $\{\mathbf{v}_t\}_{t=1,\dots,T}$, only the noisy observation $\{\mathbf{z}_t\}_{t=1,\dots,T}$ can be observed. That is,

$$\mathbf{z}_t = \mathbf{y}_t + \mathbf{v}_t. \tag{1}$$

The aim of filtering is to estimate the true signal process, denoted by $\hat{\mathbf{y}}_t$, from the noisy observed sequence $\mathbf{z}_t$ for each time t. Further, the backward filtering of $\mathbf{z}_t$ to obtain the estimates $\{\hat{\mathbf{y}}_{t'}\}_{t'<t}$ is called smoothing [1]. If the true signal process $\{\mathbf{y}_t\}$ can be well modeled by a SVH-ARMA model, both the filtering and smoothing of the noisy signal can be realized by the extended Kalman filtering algorithm, which will be discussed in Section 3.

2.2. *Prediction*

First, let us consider the problem of one step ahead prediction of time series. Assuming that we have obtained the SVH-ARMA model parameter estimates $\{\hat{\Theta}\}$

based on the already-known sequence $Y_t = [\mathbf{y}_1, \ldots, \mathbf{y}_t]_{t=1,\ldots,T}$, the objective of one step ahead prediction is to predict the value $\mathbf{y}_{t+1}$ based on the sequence Y_t [2], [9], [19]. There are L different conditional forecasts associated with the L possible states at time $t+1$. The unconditional forecast based on actual observable variables is in the form

$$\begin{aligned}
E(\mathbf{y}_{t+1}|Y_t, \hat{\Theta}) &= \int \mathbf{y}_{t+1} \left\{ \sum_{j=1}^{L} P(\mathbf{y}_{t+1}, x_{t+1}|Y_t, \hat{\Theta}) \right\} d\mathbf{y}_{t+1} \\
&= \int \mathbf{y}_{t+1} \left\{ \sum_{j=1}^{L} [u(\mathbf{y}_{t+1}|x_{t+1} = j, Y_t, \hat{\Theta}) P(x_{t+1} = j|Y_t, \hat{\Theta})] \right\} d\mathbf{y}_{t+1} \\
&= \sum_{j=1}^{L} P(x_{t+1} = j|Y_t, \hat{\Theta}) \int \mathbf{y}_{t+1} u(\mathbf{y}_{t+1}|x_{t+1} = j, Y_t, \hat{\Theta})\, d\mathbf{y}_{t+1} \\
&= \sum_{j=1}^{L} P(x_{t+1} = j|Y_t, \hat{\Theta}) E(\mathbf{y}_{t+1}|x_{t+1} = j, Y_t, \hat{\Theta}). \qquad (2)
\end{aligned}$$

According to the property of first-order Markov chains, the one step ahead prediction of the state probability is

$$P(x_{t+1} = j|Y_t, \hat{\Theta}) = \sum_{i=1}^{L} d_{ij} P(x_t = i|Y_t, \hat{\Theta}), \qquad j = 1, \ldots, L. \qquad (3)$$

Recalling the forward inference equation (24) in [22], we know that this probability can be estimated with $\beta_{t+1} = [P(x_{t+1} = 1|Y_t, \hat{\Theta}), \ldots, P(x_{t+1} = L|Y_t, \hat{\Theta})]'$.

It is straightforward to observe that

$$E(\mathbf{y}_{t+1}|x_{t+1} = j, Y_t, \hat{\Theta}) = \hat{\mu}(j) + \hat{C}(j)\Upsilon_t' + h(\Upsilon_t; \hat{W}^s_{mn}(j)). \qquad (4)$$

So the final one step ahead prediction is

$$\begin{aligned}
\hat{\mathbf{y}}_{t+1} &= \sum_{j=1}^{L} \hat{\beta}_{t+1}(j)(\hat{\mu}(j) + \hat{C}(j)\Upsilon_t' + h(\Upsilon_t; \hat{W}^s_{mn}(j))) \\
&= \sum_{j=1}^{L} \hat{\beta}_{t+1}(j)\hat{\mu}(j) + \sum_{j=1}^{L} (\hat{\beta}_{t+1}(j)\hat{C}(j))\Upsilon_t' \\
&\quad + \sum_{j=1}^{L} \hat{\beta}_{t+1}(j) h(\Upsilon_t; \hat{W}^s_{mn}(j)). \qquad (5)
\end{aligned}$$

Based on this one step ahead prediction, the multiple step ahead prediction can also be gained. We will revisit this topic in Section 4.

Remark 1. The number of states L could be determined according to the a priori knowledge of the possible stationary regimes of a nonstationary process by assigning one state to a regime. If too little a priori knowledge is available, we could estimate L by observing the time-varying regimes of the nonstationary process. Taking a large state number can track the nonstationarity well but will increase the complexity of the model and the computational load of parameter estimation. On the other hand, taking a small state number may keep the model parsimonious but is sometimes not sufficient to represent the changing of a nonstationary process. In extensive simulations we find that the performance of filtering or prediction is not so sensitive to the selection of the state number. Commonly, taking the state number as $L = 3 \sim 5$ is sufficient for most applications in practice.

2.3. Control

It is natural to extend our SVH-ARMA model to the SVH-ARMAX model, which potentially has wide applications in controls [9].

The SVH-ARMAX model can be written as follows:

$$\mathbf{y}_t = \mu(i) + \sum_{k=1}^{p} A_k(i)\mathbf{y}_{t-k} + \sum_{k=1}^{\tau} \aleph_k(i)\zeta_{t-k} + \sum_{k=1}^{q} B_k(i)\,\mathbf{e}_{t-k}$$
$$+h(\mathbf{y}_{t-1}, \ldots, \mathbf{y}_{t-p}, \zeta_{t-1}, \ldots, \zeta_{t-\tau}, \mathbf{e}_{t-1}, \ldots, \mathbf{e}_{t-q};\, W_{mn}^{s}(i)) + \mathbf{e}_t, \quad (6)$$

where $\{\zeta_t\}_{t=1,\cdot,T}$ is the exogenous variable input and $\{\aleph_k(i)\}_{k=1,\ldots,\tau;i=1,\ldots,L}$ is the exogenous variable coefficient matrix. The remaining parameters are the same as those of the SVH-ARMA model. If we take $\{\zeta_t\}_{t=1,\ldots,T}$ as the control input vector and $\{\mathbf{y}_t\}_{t=1,\ldots,T}$ as the control output vector, a linear and nonlinear mapping is readily founded.

Introduce a new model coefficient matrix as

$$C(i) = [A_1(i), \ldots, A_p(i), \aleph_1(i), \ldots, \aleph_\tau(i), B_1(i), \ldots, B_q(i)], \quad (7)$$

and a new model input vector as

$$\Upsilon_t = [\mathbf{y}'_{t-1}, \ldots, \mathbf{y}'_{t-p}, \zeta'_{t-1}, \ldots, \zeta'_{t-\tau}, \mathbf{e}'_{t-1}, \ldots, \mathbf{e}'_{t-q}]. \quad (8)$$

Then model (6) is equivalent to SVH-ARMA model (10) of [22], which can be identified using the techniques introduced in Section 3 of [22].

In the rest of the paper, we focus on some applications of the SVH-ARMA model in filtering and prediction. Applications of the SVH-ARMAX model in controls will be considered elsewhere.

3. Extended Kalman filtering for signal enhancement based on SVH-ARMA model

In the time domain, if a nonstationary and nonlinear (scalar or vector) process $\{\mathbf{y}_t\}_{t=1,\ldots,T}$ is contaminated by a Gaussian white noise $\{\mathbf{v}_t\}_{t=1,\ldots,T}$, the observed noisy sequence $\{\mathbf{z}_t\}_{t=1,\dots,T}$ becomes

$$\mathbf{z}_t = \mathbf{y}_t + \mathbf{v}_t. \tag{9}$$

In this paper, to simplify the derivation of the equivalent state-space model below and reduce the computational complexity of the filtering algorithm, we assume that the true signal process $\{\mathbf{y}_t\}_{t=1,\dots,T}$ can be modeled well by an SVH-AR model (a degenerated SVH-ARMA model) as

$$\mathbf{y}_t = \mu(i) + \sum_{k=1}^{p} A_k(i)\mathbf{y}_{t-k} + h(\mathbf{y}_{t-1}, \dots, \mathbf{y}_{t-p};\ W_{mn}^s(i)) + \mathbf{e}_t. \tag{10}$$

Then, for a certain state $x_t = i$ at time t, equations (9) and (10) can be converted to a Markov chain state-dependent vector state-space model [1] as

$$\begin{cases} \Phi_{t+1} = D(i)\Phi_t + \mathbf{N}(\Phi_t; i) + \check{\mu}(i) + F\mathbf{e}_t \\ \quad \mathbf{z}_t = G\Phi_t + \mathbf{v}_t, \end{cases} \tag{11}$$

where

$$\begin{aligned} D(i) &= \begin{bmatrix} A_1(i) & \cdots & A_{p-1}(i) & A_p(i) \\ I_J & \cdots & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & I_J & 0 \end{bmatrix}_{Jp\times Jp}, \\ \mathbf{N}(\Phi_t; i) &= [\,h(\Phi_t; W_{mn}^s(i))' \quad 0 \quad \cdots \quad 0\,]', \\ \Phi_t &= [\,\mathbf{y}_t' \quad \cdots \quad \mathbf{y}_{t-p+1}'\,]', \\ \check{\mu}(i) &= [\,\mu(i)' \quad 0 \quad \cdots \quad 0\,]', \\ F &= [\,I \quad 0 \quad \cdots \quad 0\,]', \\ G &= [\,I \quad 0 \quad \cdots \quad 0\,]. \end{aligned} \tag{12}$$

Expand the nonlinear functions $\mathbf{N}(\Phi_t; i)$ in a Taylor series about the conditional mean vector $\hat{\Phi}_t$ as

$$\mathbf{N}(\Phi_t; i) = \mathbf{N}(\hat{\Phi}_t; i) + \nabla\mathbf{N}(\hat{\Phi}_t; i)(\Phi_t - \hat{\Phi}_t) + \cdots, \tag{13}$$

where

$$\nabla\mathbf{N}(\hat{\Phi}_t; i) = \begin{bmatrix} h_1^d(i) & h_2^d(i) & \cdots & h_p^d(i) \\ 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix}_{Jp\times Jp}, \tag{14}$$

and

$$h_j^d(i) = \left.\frac{\partial h}{\partial \Phi_t^j}\right|_{(\hat{\Phi}_t^j; i)}, \qquad j = 1, \dots, p\ (\Phi_t^j \text{ is the } j\text{th element of vector } \Phi_t). \tag{15}$$

Neglecting higher order terms and assuming knowledge of Φ_t and $\Phi_{t,t-1}$

enable us to approximate the signal model (11). Then equation (11) can be rewritten as a linear state-space equation:

$$\begin{cases} \Phi_{t+1} = [D(i) + \nabla \mathbf{N}(\hat{\Phi}_t; i)]\Phi_t + [\breve{\mu}(i) + \mathbf{N}(\hat{\Phi}_t; i) - \nabla \mathbf{N}(\hat{\Phi}_t; i)\hat{\Phi}_t] + F\mathbf{e}_t \\ \quad \mathbf{z}_t = G\Phi_t + \mathbf{v}_t \end{cases} \tag{16}$$

Hence, the extended Kalman filtering (EKF) update equations [1] become

$$\begin{aligned} \hat{\Phi}_t &= \hat{\Phi}_{t,t-1} + \Xi_t(\mathbf{z}_t - G\hat{\Phi}_{t,t-1}), \\ \hat{\Phi}_{t,t-1} &= \mathbf{N}(\hat{\Phi}_{t-1}; i) + \breve{\mu}(i), \\ \Xi_t &= P_{t,t-1}G'(GP_{t,t-1}G' + \Psi_t)^{-1} \qquad (\Psi_t = E(\mathbf{v}_t\mathbf{v}_t{}')), \\ P_{t,t} &= (I - \Xi_t G)P_{t,t-1}, \\ P_{t,t-1} &= (D(i) + \nabla \mathbf{N}(\hat{\Phi}_{t-1}; i))P_{t-1,t-1}(D(i) + \nabla \mathbf{N}(\hat{\Phi}_{t-1}; i))' \\ &\quad + F\Omega_{t-1}F' \qquad (\Omega_t = E(\mathbf{e}_t\mathbf{e}_t{}')). \end{aligned} \tag{17}$$

Here, $\hat{\Phi}_{t,t-1}$ is an a priori estimation of Φ_t, $\hat{\Phi}_t$ is an a posteriori estimation of Φ_t, Ξ_t is the optimal filtering gain, $P_{t,t-1}$ is the estimated covariance matrix of $\hat{\Phi}_{t,t-1}$, and P_t is the estimated covariance matrix of $\hat{\Phi}_t$. Denote $Z(t) = [\mathbf{z}_1, \ldots, \mathbf{z}_t]$. The filtered output at state i and time t is

$$\hat{\mathbf{y}}_t(x(t) = i, Z(t)) = G\hat{\Phi}_t. \tag{18}$$

So, the optimum estimate of $\mathbf{y}_t$ is

$$\hat{\mathbf{y}}_t = E(\mathbf{y}_t|Z(t)) = \sum_{i=1}^{M} \hat{\mathbf{y}}_t(x_t = i, Z(t))P(x_t = i|Z(t)). \tag{19}$$

According to the Bayes rules, the weighting factor can be written as

$$P(x_t = i|Z(t)) = \frac{P(\mathbf{z}_t|x_t = i, Z(t-1))P(x_t = i|Z(t-1))}{P(\mathbf{z}_t|Z(t-1))}, \tag{20}$$

where

$$P(\mathbf{z}_t|x_t = i, Z(t-1)) = g(\mathbf{z}_t, \hat{\Phi}_{t,t-1}, GP_{t,t-1}G' + \Psi_t), \tag{21}$$

$$P(x_t = i|Z(t-1)) = \sum_{j=1}^{M} d_{ji}P(x_{t-1} = j|Z(t-1)), \tag{22}$$

and $P(\mathbf{z}_t|Z(t-1))$ is a constant that is irrelevant to state i.

Substituting (21) and (22) into equation (20), $P(x_t = i|Z(t))$ can be estimated recursively except for a constant $P(\mathbf{z}_t|Z(t-1))$. To ensure that

$$\sum_{i=1}^{M} P(x_t = i|Z(t)) = 1, \tag{23}$$

we can rescale $P(x_t = i|Z(t))$ as

$$\bar{P}(x_t = i|Z(t)) = P(x_t = i|Z(t)) \Big/ \sum_{i=1}^{M} P(x_t = i|Z(t)). \tag{24}$$

Once $\hat{y}_t(x_t = i, Z(t))$ (18) and $\bar{P}(x_t = i|Z(t))$ (24) are obtained separately, we can obtain the optimal filtering estimate $\{\hat{y}_t\}$ through equation (19).

The smoothing problem (fixed-lag or fixed-point smoothing) based on the same state-space model (16) for each state i [9] is similar to the filtering problem just discussed. Taking the same Bayesian inference procedure as (18)–(24), the optimal smoothing to the noisy signal can also be obtained. The details are omitted here to save space.

Remark 2. The proposed EKF algorithm is essentially signal independent because it can automatically and adaptively represent and track arbitrary nonstationary and nonlinear processes. Thus, it can provide good performance when applied to various noisy signals such as noisy images, noisy speech, noisy electrocardiograms (ECGs), and noisy seismic signals for enhancement. In this sense, the proposed filtering algorithm can be used as a unified and widely suited signal enhancement method.

4. Long correlation time series nonlinear prediction based on SVH-ARMA model and dyadic wavelet transform

In this section, a scalar time series is decomposed into several scale sequences by using a dyadic wavelet transform (DWT) [13], and then a vector time series can be constructed by grouping along the scales. The SVH-ARMA model is used for this vector time series modeling and prediction. The final forecasting to the original scalar time series can also be obtained through the inverse wavelet transform. It is verified by simulations that the accuracy of the final prediction results using this method is superior than predicting the original time series directly.

First, let us briefly introduce the theory of wavelet analysis [13]. Given a continuous function ψ in L^2, if ψ satisfies an admissibility condition, ψ is named as a basic wavelet or mother wavelet. Denote $\psi_s(t) = \frac{1}{s}\psi(\frac{t}{s})$. Then the wavelet transform of function $f(x)$ at scale s and location u is defined as

$$W_s f(u) = f * \psi_s(u) = \int_{-\infty}^{\infty} f(x)\psi_s(x-u)\,dx. \tag{25}$$

We can discretize scales according to the power of 2; that is, we can set $s = 2^j$. Then

$$W_{2^j} f(u) = f * \psi_{2^j}(u) = 2^{-j}\int_{-\infty}^{\infty} f(x)\psi\left(\frac{x-u}{2^j}\right)dx \tag{26}$$

is called the *dyadic wavelet transform* (*DWT*). Conversely, there exists a reconstructed wavelet χ, and the inverse wavelet transform can be used for perfectly reconstructing $f(x)$:

$$f(x) = W_{2^j} f * \chi_{2^j}(x) = 2^j \int_{-\infty}^{\infty} W_{2^j} f(u)\chi\left(\frac{x-u}{2^j}\right)du. \tag{27}$$

Mallat and Zhong have demonstrated the relationship between multiresolution analysis and wavelet transform [13]. If a signal is decomposed at a finite number of scales, the signals in smaller scales represent high-frequency components, and the corresponding resolutions in the time domain are high, whereas the signals in bigger scales represent low-frequency components, and the resolutions in the frequency domain are high.

Assume that the maximum scale of decomposition is J. Given a scaling function ϕ, we can obtain a wavelet function ψ and a reconstruction function χ. Define a smooth operator S as

$$S_{2^j} f(u) = f * \phi_{2^j}(u) = 2^{-j} \int_{-\infty}^{\infty} f(x)\phi\left(\frac{x-u}{2^j}\right) dx, \tag{28}$$

and a complementary operator W as in equation (26). The smoothed signal $S_{2^j} f(u)$ represents the signal's basic changing tendency at scale 2^j, and $W_{2^j} f(u)$ represents the detail elements at scale 2^j. Some high-frequency elements of $f(x)$ are lost in $S_{2^J} f(u)$, but they can be compensated by $\{W_{2^j} f(u)\}_{j=1,\dots,J}$.

A DWT fast algorithm is presented in [13], which is used for the wavelet decomposition of scalar time series in this paper. The mother wavelet we use is the first-order derivative of the cubic spline function, and its coefficients are also given in [13].

4.1. *Vector time series case*

For a one step ahead prediction, given a vector time series $\{\mathbf{y}_t | t = 1, \dots, T\}$, we need to predict the output $\mathbf{y}_t$ at time $T + 1$. The already-known sequence $\{\mathbf{y}_t | t = 1, \dots, T\}$ may be used for state and model parameter estimation. Then $\mathbf{y}_{T+1}$ can be predicted based on these estimated parameters $\{\hat{\pi}, \hat{D}, \hat{\mu}(i), \hat{C}(i), \hat{W}^s_{mn}(i)\}$ and the "past" input vector $\hat{\Upsilon}_{T+1} = [\mathbf{y}_T, \dots, \mathbf{y}_{T-p+1}, \hat{\mathbf{e}}_T, \dots, \hat{\mathbf{e}}_{T-q+1}]$ as follows:

$$\hat{\beta}_{T+1} = \hat{D}\hat{\alpha}_T, \tag{29}$$

$$\begin{aligned} \hat{\mathbf{y}}_{T+1} = {} & \sum_{i=1}^{L} (\hat{\beta}_{T+1}(i)\hat{\mu}(i)) + \sum_{i=1}^{L} (\hat{\beta}_{T+1}(i)\hat{C}(i))\hat{\Upsilon}'_{T+1} \\ & + \sum_{i=1}^{L} (\hat{\beta}_{T+1}(i) h(\hat{\Upsilon}_{T+1}; \hat{W}^s_{mn}(i))). \end{aligned} \tag{30}$$

Here, (29) represents the one step ahead prediction of Markov chain state distribution and (30) represents the one step ahead prediction of a future time series value based on the estimated SVH-ARMA model. Obviously, the one step ahead prediction value $\hat{\mathbf{y}}_{T+1}$ (30) is the weighted average of L conditional forecasts associated with L possible states at time $T + 1$, where the weighting coefficients are the state distribution at time $T + 1$ predicted by (29). For a nonstationary process, the changing of regimes will result in the varying of the state distribution.

Thus, the final prediction value will be affected by the nonstationary varying of regimes of the process. Consequently, the predicted value looks more like the time series in the nearer regime than the time series in the regime far away from it. In this sense, the prediction based on the SVH-ARMA model is self-tuning and adaptive to the varying of the nonstationary processes.

In the case of the multiple ($K > 1$) ahead prediction, $\mathbf{y}_{T+1}, \ldots, \mathbf{y}_{T+K}$ can be predicted in a recursive procedure. Once we obtain the one step prediction value $\hat{\mathbf{y}}_{T+1}$, we take it in the place of the true value of $\mathbf{y}_{T+1}$ and make the next one step prediction based on $\{\mathbf{y}_1, \ldots, \mathbf{y}_t, \hat{\mathbf{y}}_{T+1}\}$ to obtain $\hat{\mathbf{y}}_{T+2}$. We then take it instead of the value of $\mathbf{y}_{T+2}$ and make the next prediction, ..., until we attain the $\hat{\mathbf{y}}_{T+K}$.

Remark 3. It has been reported that neither the linear prediction nor prediction with a compressive nonlinear function alone can produce long-range temporal correlation in the output of the model. However, a joint prediction with linear and compressive nonlinear functions could provide sufficient long-range prediction [7]. In our SVH-ARMA model, it includes two parts—the linear ARMA part and the nonlinear ARMA part. The nonlinear ARMA part is realized by MLFF neural networks. Consequently, if the operating function of the output layer of neural networks is set as a compressive function such as "logsig" or "tansig" etc., the constructed SVH-ARMA model can be used for long-range prediction.

4.2. Scalar time series case

The wavelet scale space is composed of wavelet transform coefficients $W_{2^j} f$ ($j = 1, \ldots, J$) and smooth coefficients $S_{2^J} f$, written as $\{W_{2^j} f(t), S_{2^J} f(t) | j = 1, \ldots, J; t = 1, \ldots, T\}$. Obviously, we can convert a scalar time series to a vector time series in wavelet scale space and then make multiresolution analysis and modeling for this vector process.

To fulfill the prediction of a scalar time series $\theta(t)$ in the wavelet scales space, we transform it by the undecimated DWT to J scales and then group along the scales to obtain a series of vector-valued signals,

$$\mathbf{y}_t = [W^d_{2^1}\theta(t), \ldots, W^d_{2^J}\theta(t), S^d_{2^J}\theta(t)]'. \tag{31}$$

Each $\mathbf{y}_t$ is a column vector of size $(J + 1) \times 1$. Here, $\{W^d_{2^j}\theta(t)\}$ is the discrete sampling of $\{W_{2^j}\theta\}$ at integer instants, and $\{S^d_{2^J}\theta(t)\}$ is the discrete sampling of $\{S_{2^J}\theta\}$ at integer instants.

When the vector time series $\{\mathbf{y}_t\}$ is obtained, it can be modeled by an SVH-ARMA model. The sequence $\{\mathbf{y}_t | t = 1, \ldots, T\}$ can be taken for the SVH-ARMA model parameter estimation and then one step ahead prediction with (29) and (30) can be made. For multiple step ahead prediction, we can predict $\mathbf{y}_{T+1}, \ldots, \mathbf{y}_{T+K}$, $K \geq 1$, based on $\mathbf{y}_T, \ldots, \mathbf{y}_{T+K-1}$ step by step. By applying the inverse wavelet transform to the vector prediction values $\hat{\mathbf{y}}_T, \ldots, \hat{\mathbf{y}}_{T+K}$, the multiple step ahead prediction of the scalar time series $\{\theta(t)\}$, i.e., $\hat{\theta}(T), \ldots, \hat{\theta}(T + K)$ can be obtained. Because $\{\mathbf{y}_t\}$ represents the multiscale characteristics of the scalar process

$\{\theta(t)\}$, predicting $\{\mathbf{y}_t\}$ instead of predicting $\{\theta(t)\}$ directly would attain a more accurate prediction.

Remark 4. The prediction by the SVH-ARMA model is optimal in one step ahead prediction but not optimal in multiple step ahead prediction. To further increase the prediction accuracy, we may combine the SVH-ARMA model with another neural network for the final K step prediction. To each $\{\mathbf{y}_t | t = \max(p, q) + 1, \ldots, T - K\}$, we can obtain K step predicted values $\{\hat{\mathbf{y}}_{t+1}, \ldots, \hat{\mathbf{y}}_{t+K}\}$ by the SVH-ARMA model. Then we take $\{\hat{\mathbf{y}}_{t+1}, \ldots, \hat{\mathbf{y}}_{t+K}\}$ as the input and $\{\mathbf{y}_{t+1}, \ldots, \mathbf{y}_{t+K}\}$ $(\max(p, q) + 1 \leq t \leq T - K)$ as the target output to train a new MLFF neural network (named the K step prediction neural network). Once the training of the MLFF neural network converges, it can be used for the final K step prediction to $\theta(T), \ldots, \theta(T + K)$. We just need to input $\hat{\mathbf{y}}_T, \ldots, \hat{\mathbf{y}}_{T+K}$ (obtained by prediction with the SVH-ARMA model) to the K step prediction neural network, and then the output of the neural networks will produce the more accurate K step prediction.

5. Simulation results

5.1. Performance of model representation and model validation

In this simulation, we take a two-dimensional (2D) vector time series as an example to demonstrate the representation ability of our SVH-ARMA model.

Produce a 2D vector linear ARMA(2, 2) process $\{\mathbf{y}_t^L = [\mathbf{y}_t^L(1)\, \mathbf{y}_t^L(2)]', t = 1, \ldots, T\}$ as

$$\mathbf{y}_t^L = \sum_{k=1}^{2} A_k \mathbf{y}_{t-k}^L + \sum_{k=1}^{2} B_k \mathbf{e}_{t-k} + \mathbf{e}_t, \tag{32}$$

where

$$A_1 = \begin{bmatrix} 0.15 & 0 \\ -0.23 & 0.74 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 0.51 & 0.28 \\ 0 & 0 \end{bmatrix},$$
$$B_1 = \begin{bmatrix} 1.91 & -0.02 \\ 0.61 & -1.34 \end{bmatrix}, \quad B_2 = \begin{bmatrix} -0.53 & 1.64 \\ 0 & 0 \end{bmatrix}, \tag{33}$$

and the covariance matrix of the zero-mean vector driving noise sequence $\{\mathbf{e}_t = [\mathbf{e}_t(1)\, \mathbf{e}_t(2)]', t = 1, \ldots, T\}$ is

$$\operatorname{cov}(\mathbf{e}) = \begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix}, \tag{34}$$

where cov represents a covariance matrix.

Produce a 2D dynamic vector nonlinear process $\{\mathbf{y}_t^N = [\mathbf{y}_t^N(1)\, \mathbf{y}_t^N(2)]', t =$

$1, \ldots, T\}$ according to the recursive equation

$$\left\{\begin{array}{l} \begin{bmatrix} \tilde{\mathbf{y}}_{t+1}^N(1) \\ \tilde{\mathbf{y}}_{t+1}^N(2) \end{bmatrix} = 4 \begin{bmatrix} \tilde{\mathbf{y}}_t^N(1) & 0 \\ 0 & \tilde{\mathbf{y}}_t^N(2) \end{bmatrix} \begin{bmatrix} 1-\tilde{\mathbf{y}}_t^N(1) & 0 \\ 0 & 1-\tilde{\mathbf{y}}_t^N(2) \end{bmatrix}, \\ \qquad \mathbf{y}_t^N = \tilde{\mathbf{y}}_t^N + \mathbf{e}_t \end{array}\right. \tag{35}$$

and the initial values are $\tilde{\mathbf{y}}_1^N(1) = 0.3$ and $\tilde{\mathbf{y}}_1^N(2) = 0.6$, respectively.

Then a process having both linear and nonlinear components can be constructed as

$$\mathbf{y}_t = \varrho\, \mathbf{y}_t^L + (1-\varrho)\, \mathbf{y}_t^N, \tag{36}$$

where $\varrho < 1$ is a constant factor that balances the ratio of the amplitude between the linear component and the nonlinear component.

One realization of the vector time series (the length of the data is $T = 200$) (36) is shown in Figure 1. Prediction error signals by the SVH-ARMA model are also shown in the same figure corresponding to each trace. As can be seen, the model predictive outputs coincide with the true signal very well. For this example, we choose the SVH-ARMA model orders for identification as follows: the linear ARMA part order $p = 2$, $q = 2$, the nonlinear ARMA part order $p_n = 5$, $q_n = 2$, with a three-layer MLFF neural networks, the size of neural networks being $(p_n + q_n) \times 2$, $p_n \times 2$, and 2 for the input layer, hidden layer, and output layer, respectively, and the state number $L = 1$. The prediction error signal $\{\mathbf{e}_t\}$ can be obtained when the SVH-ARMA is set up (i.e., the model parameters have been estimated based on the training sequence) as

$$\hat{\mathbf{e}}_t = \hat{\mathbf{y}}_t - \sum_{i=1}^{L}(\hat{\gamma}_t(i)\hat{\mu}(i)) + \sum_{i=1}^{L}(\hat{\gamma}_t(i)\hat{C}(i))\hat{\Upsilon}_t' + \sum_{i=1}^{L}(\hat{\gamma}_t(i)h(\hat{\Upsilon}_t; \hat{W}_{mn}^s(i))). \tag{37}$$

Model validation can be implemented by using the autocorrelation function (ACF) or cross-correlation function (XCF) of the prediction error signals, which are shown in Figure 2. Here, the ACF or XCF is defined as follows:

$$\tilde{\Phi}_{\mathbf{e}^i\mathbf{e}^i}(k) = \frac{\sum_{l=1}^{T-k} \mathbf{e}_l^i \mathbf{e}_{l+k}^i}{T-k}, \qquad i = 1, 2;\ k = 1, \ldots, 30 \tag{38}$$

$$\tilde{\Phi}_{\mathbf{e}^i\mathbf{e}^j}(k) = \frac{\sum_{l=1}^{T-k} \mathbf{e}_l^i \mathbf{e}_{l+k}^j}{T-k}, \qquad i, j = 1, 2;\ i \neq j,\ k = 1, \ldots, 30. \tag{39}$$

Figure 2 shows that each trace of prediction error signal is uncorrelated, and trace 1 and 2 are also uncorrelated. Figures 1 and 2 verify that our model representation is efficient and sufficient [17].

In order to compare the representation ability of the linear ARMA model, the nonlinear ARMA model, and the SVH-ARMA model, we apply the three methods to the processes $\{\mathbf{y}_t^L\}$, $\{\mathbf{y}_t^N\}$, or $\{\mathbf{y}_t\}$ above separately. Their relative predictive error ratios (RPERs) are shown in Table 1. The RPER of a process $\mathbf{s}$ is defined as

$$\mathrm{RPER} = \frac{\sum_{i=1}^{2}\sum_{t=1}^{T} \hat{\mathbf{r}}_t(i)^2}{\sum_{i=1}^{2}\sum_{t=1}^{T} \mathbf{s}_t(i)^2}. \tag{40}$$

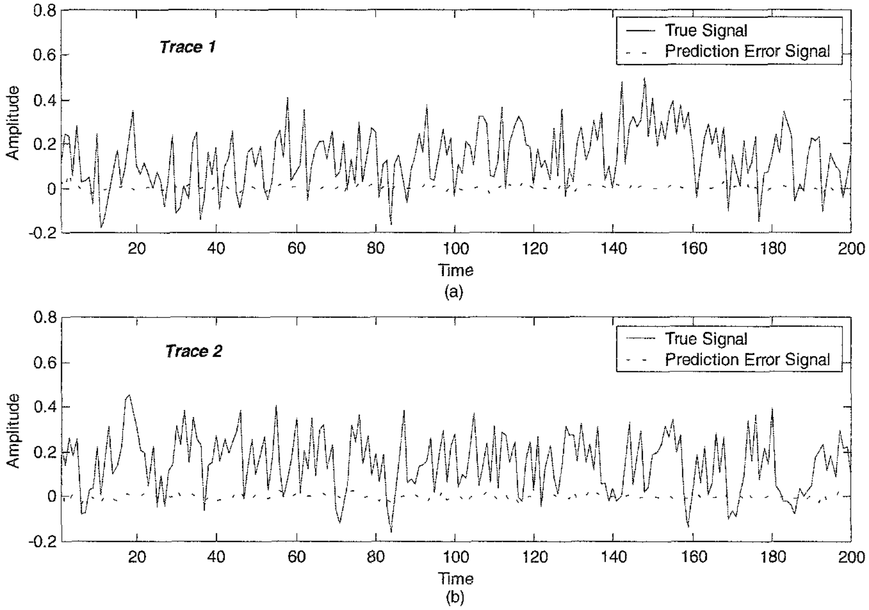

Figure 1. A bivariable time series and the hybrid ARMA model prediction error signal.

Here, $\{\hat{\mathbf{r}}_t\}$ is the prediction error signal obtained by applying one of the three models above to a process $\{\mathbf{s}_t\}$ (one of the $\{\mathbf{y}_t^L\}$, $\{\mathbf{y}_t^N\}$, and $\{\mathbf{y}_t\}$). Obviously, the smaller the RPER that is obtained by a model, the higher representing ability the model possesses. In this simulation, the SVH-ARMA model order is $p = 2$, $q = 2$, $p_n = 5$, $q_n = 2$, with the same neural network structure as above, the linear ARMA model order is $p = 2, q = 2$, which is the same as the linear ARMA part of the SVH-ARMA model, the nonlinear ARMA model order is $p_n = 5$, $q_n = 2$, which is the same as the nonlinear ARMA part of the SVH-ARMA model, and the state number is $L = 1$. For each type of process and each model, a total of 100 random signal traces are run, and the average RPER is obtained and listed in Table 1. It is clearly shown in Table 1 that the linear ARMA model can model the linear process well (assuming with correct order) but poorly represents the nonlinear process; on the other hand, the nonlinear ARMA model can model the nonlinear process well but poorly represents the linear process. It appears that the SVH-ARMA model can model all the processes very well. Obviously, if the same prediction error bounds as SVH-ARMA model need to be reached when using a nonlinear model to represent a linear process, the only way is to increase the model order significantly.

Further simulations have also been conducted which show that to obtain the same prediction error bounds to the hybrid process $\{\mathbf{y}_t\}$ above, the nonlinear ARMA model needs at least a two times higher order than the SVH-ARMA

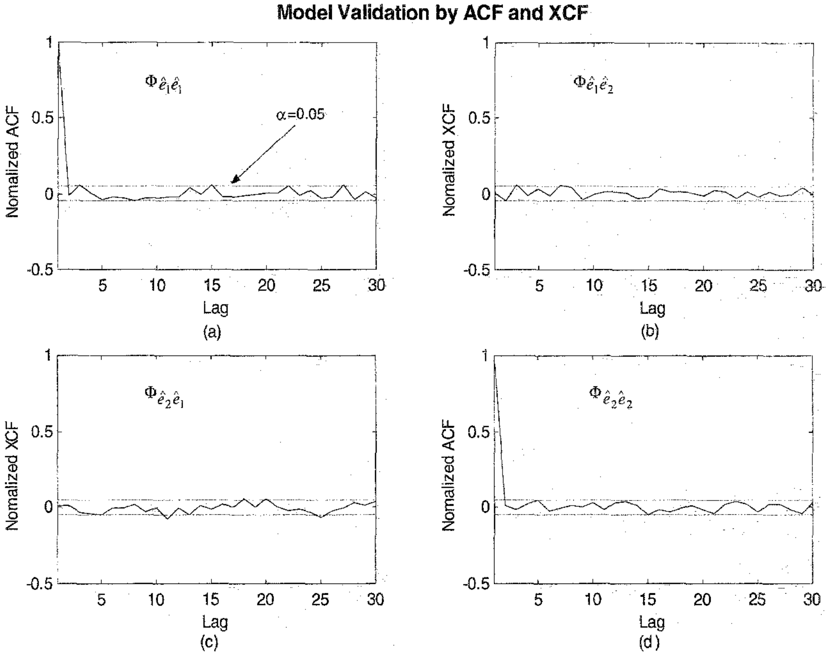

Figure 2. Model validation by autocorrelation and cross-correlation.

Table 1. Comparison of the representation ability of three models

	Average RPER values		
Model Time series	Linear ARMA model	Nonlinear AR model using neural network	SHV-ARMA model
Bivariable linear process y^L	0.0275	0.165	0.0752
Bivariable nonlinear process y^N	0.315	0.0070	0.0033
Bivariable linear/nonlinear process y	0.341	0.135	0.0235

model. On the other hand, no matter how high an order the linear ARMA model takes, it cannot reach the same error bounds possible by using the SVH-ARMA model.

5.2. Noisy image enhancement by EKF

We apply our algorithm in Section 3 to enhance a noisy image [6]. This image is corrupted by both "salt-and-pepper" non-Gaussian pulse-like noise and Gaussian white noise. The original image can be organized as a one-dimensional (1D) scalar sequence along a predetermined path (for example, horizontal, vertical, or diagonal). To filter the scalar image sequence in the wavelet scales space domain, the DWT is used to transform the original image to J scales and then group along the scales to obtain a series of vector-valued signals $\{\mathbf{y}_t\}$. It is well known that Wiener filters can eliminate Gaussian white noise but are not effective for speckle noise, and median filters can remove pulse-like noise but are not optimal for white noise [6]. However, if an image is contaminated by different types of noises simultaneously, both Wiener filtering and median filtering fail to provide good image enhancement. The proposed model inherently can represent an arbitrary nonstationary and nonlinear process, and thus the proposed filtering algorithm can adaptively "recognize" various noises from the image sequence and eliminate all the noises effectively. In the processing of the noisy image by the proposed algorithm, the vector sequence represents the multiscale information (especially edges) of the original image in the wavelet scale space domain. State-dependent training and vector-valued linear and nonlinear ARMA modeling ensure that the model can correctly approximate and track the edge features of the original image. The adaptive EKF and Bayesian pattern (state) combination ensures optimal filtering of white noises that are not necessarily Gaussian. So our method can smooth an image while retaining most of its edge features at the same time.

In the simulation, we set the number of decomposition scales of DWT to be $J = 4$, the number of states in the SVH-AR model to be $L = 4$, the order of the linear AR part to be $p = 3$, and the order of the nonlinear AR part to be $p_n = 4$ with a three-layer MLFF neural network. (The size of the neural network is $(p_n + 1) \times (J + 1)$, $p_n \times (J + 1)$, and $J + 1$ for the input layer, hidden layer, and output layer, respectively.) In Figure 3, the results of a denoised image are shown. The noise is relatively large compared to the original image (signal-to-noise power ratio: SNR $= -0.116$ dB). The Wiener filtering eliminates the Gaussian white noise very well but cannot tackle the pulse-like (speckle) noise effectively. The median filtering eliminates the speckle noise. However, it also smears many edge features of the original image. The proposed algorithm suppresses most of the white noise and the speckle noise while retaining the clear edge features. The filtering gain (Gn) of the SNR through our algorithm is about 10.2 dB, which is higher than that for both the Wiener filtering (7.92 dB) and the median filtering

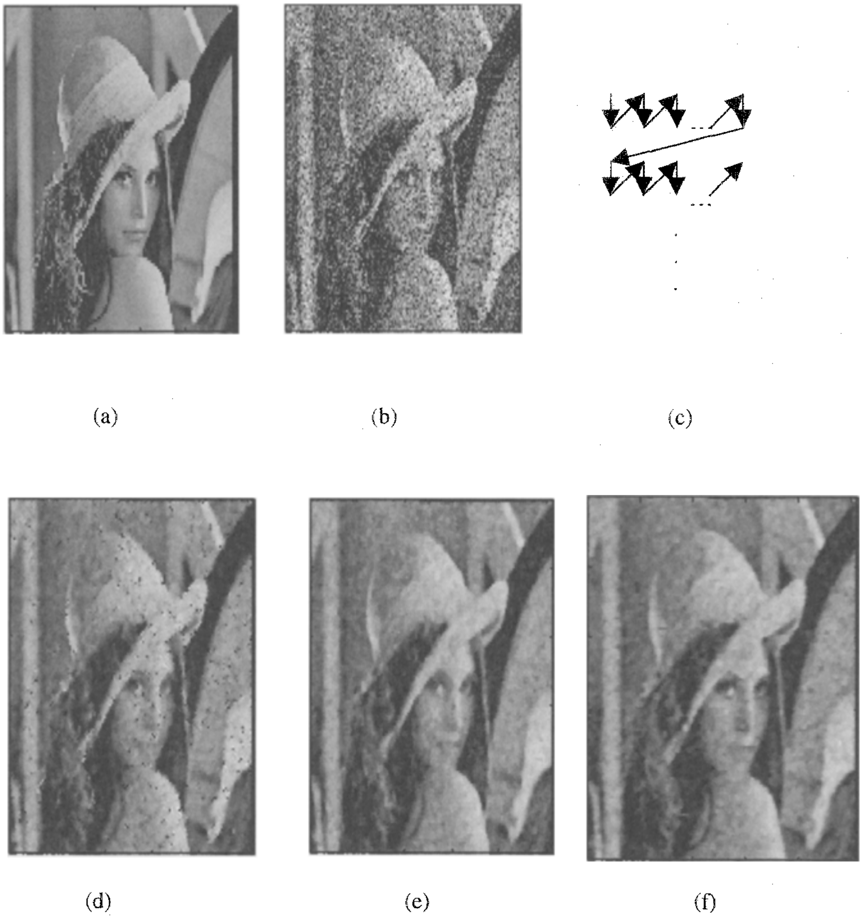

Figure 3. The results of image denoising. (a) Original image, (b) noisy image (SNR = 0.116 dB), (c) route of filtering, (d) Wiener filtering (Gn = 7.92 dB), (e) median filtering (Gn = 8.75 dB), (f) proposed filtering (Gn = 10.25 dB).

(8.75 dB). Moreover, the visual quality of our reconstructed image is superior to that obtained by the Wiener and median filtering methods.

Applying our SVH-AR model to other noisy images (contaminated by both Gaussian white noise and speckle-like noise) for denoising, the filtering gains are always about 2–3 dB higher than for Wiener filtering and median filtering for the same input SNR. This verifies that the proposed algorithm is robust and efficient. Conversely, the performance of Wiener filtering and median filtering is signal dependent and is very sensitive to the type of contaminating noises.

We point out that the image enhancement algorithms in the literature commonly are signal and case dependent (for example, [12], [21]). They are applied to specific images and/or noise types to attain good results. Different from conventional methods, the proposed algorithm is essentially signal independent. For a given noisy image, if some a priori knowledge about the original image and the noises is utilized, we believe that the proposed method could attain better results than most of the existing methods by incorporating this a priori information at the cost of a few more computations.

5.3. Stock return data prediction (vector case)

Stock return prediction plays an important role in economics. Many stock analysts and technical traders have attempted to convert their supposed forecasting prowess into trading profits in the stock markets. Recent studies have found evidence for nonstationary and nonlinear characteristics in stock returns series [3], [11], [16]. Commonly, every item of a stock sequence has its own varying tendency. At the same time, different stock time series affect each other, and there exist some potential relationships among them. Hence, it makes sense to group the multiple relevant stock sequences into a vector time series. Some major events during certain periods in history such as World War II and economic inflations would notably affect the stock changing tendency. Considering these factors, the stock time series can be modeled as a regime (state)-dependent vector hybrid linear and nonlinear ARMA (SVH-ARMA) model. As will be shown, accurate one step and multiple step ahead predictions can then be obtained based on this model even with a low model order. This will also verify the strong representing ability of our SVH-ARMA model.

We will consider the prediction of the United States stocks total returns (1871–1996). Three stock sequences will be used for illustration of the prediction. The first sequence is the composite stocks, the second is the industrials stocks, and the third is the transports stocks. We combine them all as a three-dimensional vector $\mathbf{y}_t$ and use the first 3×108 data points (1871–1978) as training sequences. The orders of the linear ARMA part are chosen as $p = 4$, $q = 2$, the orders of the nonlinear ARMA part are chosen as $p_n = 4$, $q_n = 2$, with a three-layer neural network (size: $(p_n + q_n) \times 3$, $p_n \times 3$, and 3), and the total state number is $L = 3$.

Stock return time series are obviously exponentially changing and thus are nonstationary. A common method to change an exponential time series to a stationary time series is to take a log operation and then make the first-order difference [2]. This approach will be adopted in this simulation. Both the one step and multiple step ahead predictions using the SVH-ARMA model are based on this transformed time series. However, even after the log difference transform, the resultant sequence is still nonstationary. This can be seen clearly from the abnormal changes of stock values during the years 1925–1935 (see Figure 4 a1–c1). Thus the state-dependent modeling is necessary. Once a prediction of the transformed time series is obtained, the inverse transforms (cumsum and exponential) can be applied to get a prediction of the original stock time series. The

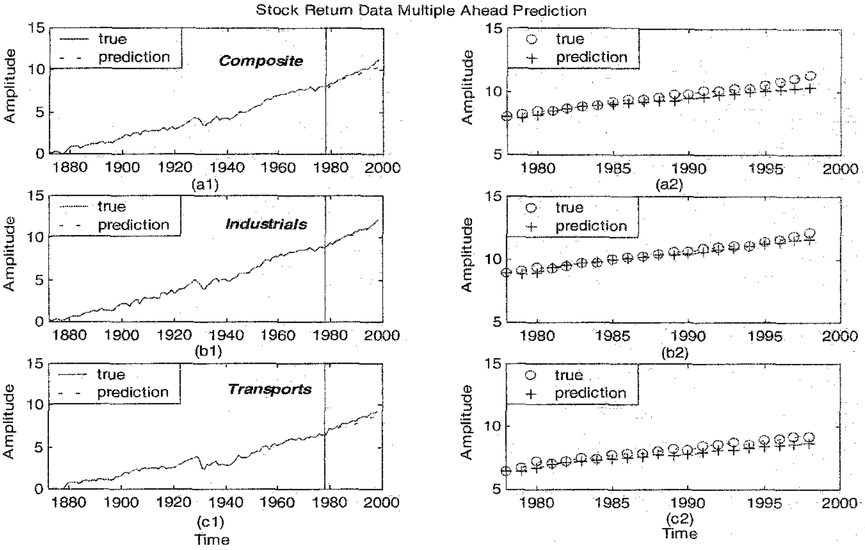

Figure 4. Three traces stock return data time series multiple step ahead prediction (20 step). Panels (a1), (b1), (c1) show the whole traces of stock return sequences (including the training sequence and multiple step ahead prediction sequence, which is separated by the vertical black line in time axis 1978). Panels (a2), (b2), (c2) show the multiple step ahead preduction traces of stock return sequences (from 1979 to 1998).

last 3 × 20 data points (1979–1998) are used for testing the accuracy of one step ahead prediction and multiple step ahead prediction.

Figure 4 presents the final results of multiple step ahead prediction (20 step) of three stock return time series (plotted in the log of the original stock sequences). It can be seen that each log sequence of the original stocks approximately changes as a linear tendency with small irregular fluctuations due to the nonlinearity and nonstationary properties of the original stock time series. Clearly, our multiple step ahead prediction using the SVH-ARMA model can track well the changing tendency after the year 1979. The nonstationary varying of stock sequences during the years 1925–1935 has a nearly negligible effect on the multiple step ahead prediction since the year 1979. This verifies the power of nonstationary modeling.

Figure 5 a1–c1 show the results of one step ahead prediction of the log difference sequence of the original stocks. The one step ahead prediction of the original stocks (plotted in the log of original stock sequences) are shown in Figure 5 a2–c2. It is seen that our model can give quite good one step ahead predictions of the original stock sequences.

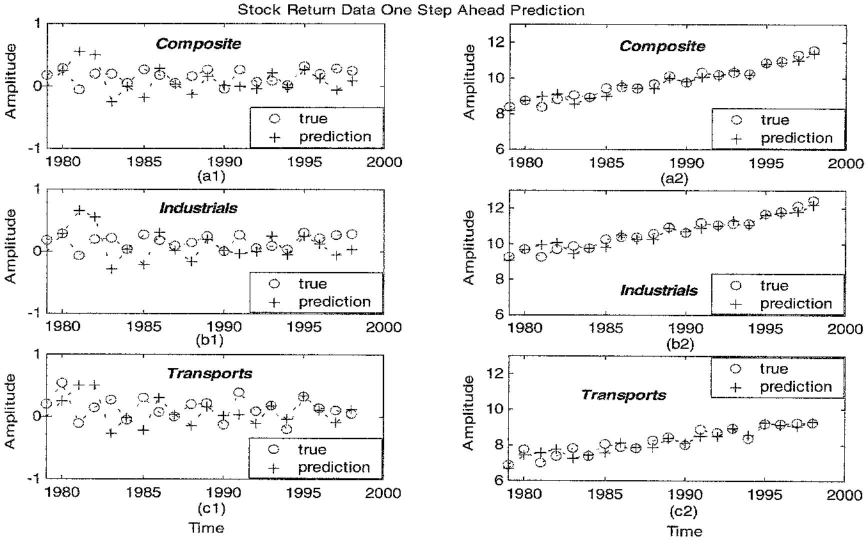

Figure 5. Three traces stock return data time series one step ahead prediction. Panels (a1), (b1), (c1) show the one step ahead prediction of log difference of stock return sequence. Panels (a2), (b2), (c2) show the one step ahead prediction of true stock return sequence.

5.4. Sunspots data prediction (scalar case)

Sunspots data is a commonly used benchmark for scalar time series prediction because of its interesting mixture of regularity and irregularity [14], [18]. It is considered as a nonstationary, medium noise and medium nonlinear time series [17, pp. 409–417]. Sunspots data display obvious cyclic phenomena in the time domain but the amplitude is time varying, which may be modeled well by our nonstationary SVH-ARMA model. In this example, we study the one step and multiple step ahead prediction ability of the SVH-ARMA model when applied to sunspots data, and compare with both a linear ARMA model [2] and a neural networks AR model [15]. It will be shown that our SVH-ARMA model can provide more accurate and confident prediction for long-range correlation data.

As demonstrated in Section 4.2, the DWT can be used here to transform the training data to multiscale data sequences and then group along scales to produce a vector sequence $\{\mathbf{y}_t\}$. Modeling $\{\mathbf{y}_t\}$ by an SVH-ARMA model, we can make vector one step and multiple step ahead predictions in the wavelet scale space domain. Applying the inverse DWT transform to these prediction values, the scalar prediction results of the original sunspots data can be obtained.

The time intervals between maxima of sunspots time series range from 7 to 15 years, and this is a guideline for SVH-ARMA model order selection. In this simulation, the number of the DWT decomposition scale is set as $J = 2$. The model orders are selected as follows: the linear ARMA part is $p = 3$, $q = 1$,

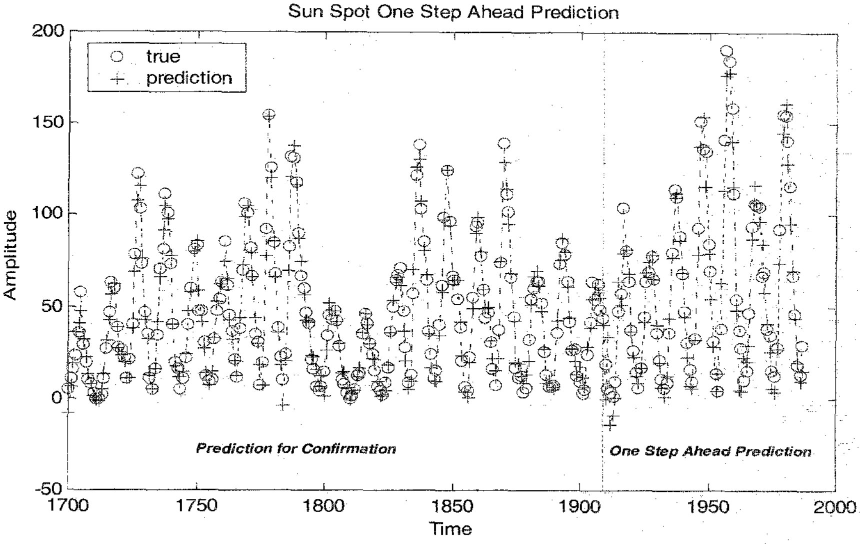

Figure 6. Sunspots one step ahead prediction.

the nonlinear ARMA part is $p_n = 10$, $q_n = 1$, with a three-layer neural network (size: $(p_n + q_n) \times (J + 1)$, $p_n \times (J + 1)$, and $J + 1$), and the state number is $L = 3$. The first 210 data points (years 1700–1909) are used for model training, and the last 78 data points (years 1910–1987) are used for prediction and testing. Figure 6 shows the results of one step ahead prediction. It verifies that the chosen model can attain quite good one step ahead prediction.

Figure 7c shows the multiple step (30 steps) ahead prediction by the SVH-ARMA model. The first 20 predicted sample values are quite close to the true sample values. However, the prediction cannot track the nonstationary changing of the time-varying amplitudes after about 20 samples. We believe that this problem is due to the unpredictability of the given data in certain nonstationary periods rather than the new method because our simulation results are much better than those obtained by conventional prediction methods. For instance, Figures 7a and b show the results of multiple step ahead prediction of the sunspots data by a linear ARMA model (order $p = 3, q = 1$) and a three-layer MLFF neural network as a nonlinear AR model (order $p_n = 10$, size of neural network: $(p_n+1)\times(J+1)$, $p_n \times (J+1)$, and $J+1$), respectively. It can be seen that the linear ARMA model makes a poor multiple step ahead prediction because it decays to a mean value very rapidly. Although the neural network AR model can predict the long-term limit cycle tendency, it cannot obtain accurate amplitude forecasting even for the first 20 predicted samples.

Figure 8 shows the three scale sequences obtained by decomposing the original sunspots data to $J = 2$ scales using the DWT. The comparison of prediction for

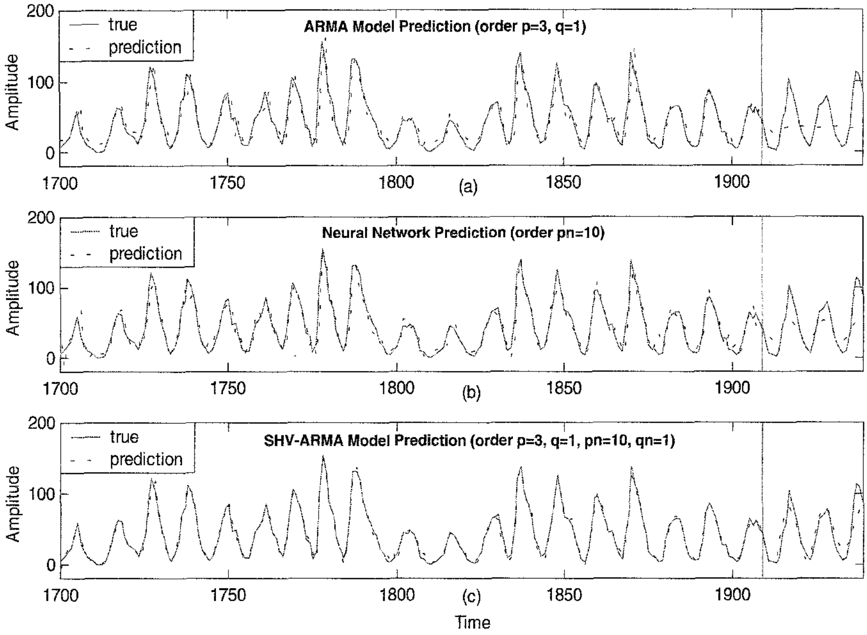

Figure 7. Sunspots multiple step ahead prediction by SHV-ARMA model (30 step).

confirmation as well as the one step ahead prediction with the true data in the wavelet scale space domain are shown. The model can attain good one step ahead prediction in every scale. However, from large scales to smaller ones, the high-frequency components of the scale sequences increase, and thus the accuracy of both modeling and prediction degenerates slightly.

6. Conclusion

In this paper, some applications of the SVH-ARMA model [22] have been presented. It has been verified by simulations that the SVH-ARMA model has strong representation ability for a general process with a low model order. The SVH-ARMA model is suitable for nonstationary noisy signal filtering and smoothing. For this, an equivalent Markov chain state-dependent state-space model has been derived, and the Bayesian rule has been employed for a posteriori state probability estimation. The extended Kalman filtering algorithm is then developed, and it can track and optimally filter both the linear and nonlinear components of an original noisy process. The SVH-ARMA model has been shown to possess good one step and multiple step ahead prediction ability due to the integration of the linear and nonlinear components. Especially, it can attain more accurate

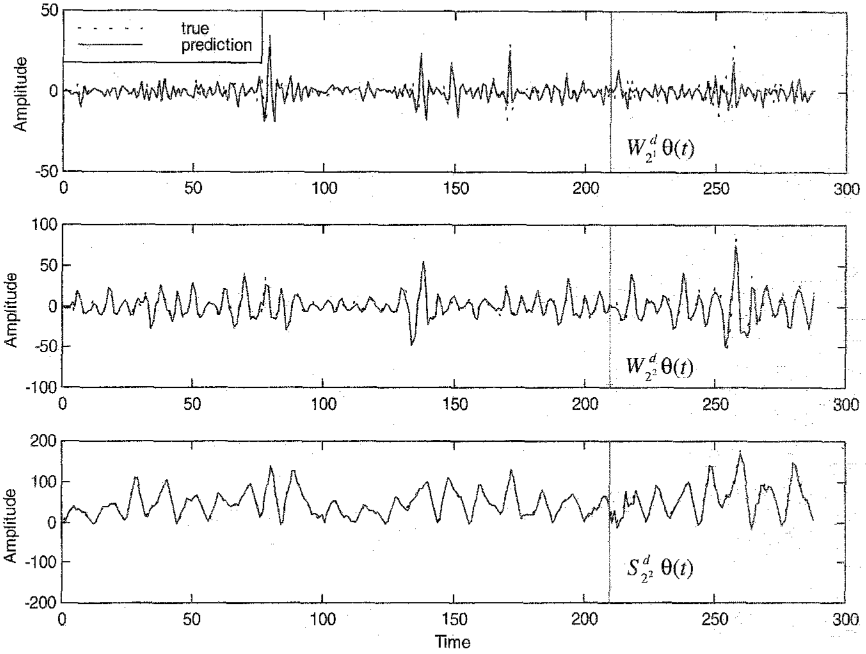

Figure 8. Three scales wavelet decomposition sequence of sunspots data and prediction in scale space.

long-range prediction than either the linear model or the nonlinear model alone. Moreover, dyadic wavelet transform has been used to transform a scalar time series to a vector time series, and the problem of the scalar time series prediction is converted to the one of a vector time series prediction in the wavelet domain. More applications of the SVH-ARMA model in signal processing, image processing, controls, and economics, etc., are expected, and will be investigated in the future.

Acknowledgments

The authors thank the anonymous reviewers, whose commenents and suggestions have greatly improved the quality of the paper.

References

[1] B. D. O. Anderson and J. B. Moore, *Optimal Filtering*, Prentice-Hall, Englewood Cliffs, NJ, 1979.

[2] P. J. Brockwell, *Introduction to Time Series and Forecasting*, Springer, New York, 1996.

[3] T. Bollerslev, Generalized autoregressive conditional heteroscedasticity, *J. Econometrics*, 21, 307–328, 1986.

[4] G. E. P. Box and G. M. Jenkins, *Time Series Analysis: Forecasting and Control*, Holden-Day, San Francisco, CA, 1976.

[5] M. Casdagli and S. Eubank (eds), *Nonlinear Modeling and Forecasting*, Addison-Wesley, Reading, MA, 1992.

[6] K. R. Castleman, *Digital Image Processing*, Prentice-Hall, Englewood Cliffs, NJ, 1996.

[7] L. Deng, K. Hassanein, and M. Elmasry, Analysis of the correlation structure for a neural predictive model with application to speech recognition, *Neural Netw.*, 7(2), 331–339, 1994.

[8] A. B. Geva, ScaleNet—multiscale neural-network architecture for time series prediction, *IEEE Trans. Neural Netw.*, 9(5), 1471–1482, 1998.

[9] G. C. Goodwin and K. S. Sin, *Adaptive Filtering Prediction and Control*, Prentice-Hall, Englewood Cliffs, NJ, 1984.

[10] J. D. Hamilton, *Time Series Analysis*, Princeton University Press, Princeton, NJ, 1994.

[11] M. J. Hinich and D. M. Patterson, Evidence of nonlinearity in daily stock returns, *J. Bus. Econ. Stat.*, 3(1), 69–77, 1985.

[12] H. Kong and L. Guan, An adaptive approach for removing impulsive noise in digital images, in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 4, pp. 2287–2290, 1996.

[13] S. Mallat and S. Zhong, Characterization of signals from multiscale edges, *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(7), 710–732, 1992.

[14] M. Mundt, W. B. Maguire, and R. P. Chase, Chaos in the sunspot cycle: analysis and prediction, *J. Geophys. Res.*, 96, 1705–1716, 1991.

[15] D. W. Patterson, *Artificial Neural Networks, Theory and Applications*, Prentice-Hall, Singapore, 1996.

[16] J. A. Scheinkman and L. Blake, Nonlinear dynamics and stock returns, *J. Business*, 62(3), 311–338, 1989.

[17] H. Tong, *Non-linear Time Series: A Dynamical System Approach*, Oxford University Press, New York, 1990.

[18] H. Tong and K. S. Lim, Threshold autoregressive, limit cycles and cyclical data, *J. R. Stat. Soc.* B, 42, 245–282, 1980.

[19] A. S. Weigend and N. A. Gershenfeld (eds), *Time Series Prediction: Forecasting the Future and Understanding the Past*, Addison-Wesley, Reading, MA, 1994.

[20] Q. Zhang and A. Benveniste, Wavelet network, *IEEE Trans. Neural Netw.*, 3(6), 889–998, 1992.

[21] Z. Z. Zhang and N. Ansari, Structure and properties of generalized adaptive neural filters for signal enhancement, *IEEE Trans. Neural Netw.*, 7(4), 857–868, 1996.

[22] Y. Zheng, Z. Lin, and D. B. H. Tay, State-dependent vector hybrid linear and nonlinear ARMA modeling: theory, Circuits Syst. Signal Process., 20, 551–574, 2001.