On-line adaptive chaotic demodulator based on radial-basis-function neural networks

Jiu-chao Feng* and Chi K. Tse†
Department of Electronic and Information Engineering, Hong Kong Polytechnic University, Hong Kong, China
(Received 3 November 1999; revised manuscript received 17 October 2000; published 17 January 2001)

Chaotic modulation is a useful technique for spread spectrum communication. In this paper, an on-line adaptive chaotic demodulator based on a radial-basis-function (RBF) neural network is proposed and designed. The demodulator is implemented by an on-line adaptive learning algorithm, which takes advantage of the good approximation capability of the RBF network and the tracking ability of the extended Kalman filter. It is demonstrated that, provided the modulating parameter varies slowly, spread spectrum signals contaminated by additive white Gaussian noise in a channel can be tracked in a time window, and the modulating parameter, which carries useful messages, can be estimated using the least-square fit. The Henon map is chosen as the chaos generator. Four test message signals, namely, square-wave, sine-wave, speech and image signals, are used to evaluate the performance. The results verify the ability of the demodulator in tracking the dynamics of the chaotic carrier as well as retrieving the message signal from a noisy channel. [S1063-651X(00)15911-2]

DOI: 10.1103/PhysRevE.63.026202 PACS number(s): 05.45.Vx, 84.35.+i

I. INTRODUCTION

Chaotic dynamical systems, characterized by quickly decaying correlation functions and broadband power spectra, provide a rich mechanism for spread spectrum (SS) signal generation and transmission. Since Yamada and Fujisaka [1,2] and Pecora and Carroll [3,4] demonstrated the synchronizability of coupled chaotic systems, there has been growing interest in making use of chaos to transmit message signals. Chaotic communication can be considered as a spread spectrum communication, which spreads a message signal into a much wider bandwidth in comparison with the message signals. The message signal is then retrieved at the receiving end by coherent or noncoherent despreading and demodulation.

A number of chaotic communication schemes have been proposed, such as *chaos masking*, *chaos shift keying*, and *chaotic modulation.* In chaos masking [5,6], a message signal with a smaller amplitude in comparison with the chaotic carriers is added to the chaotic carrier, then an identical chaotic system included at the receiver is used to ''filter'' out the message signal. This approach suffers from the disadvantage that distortion and noise introduced by the channel are indistinguishable from the signal. Furthermore, if the amplitude of the signal is too large relative to the carrier's, synchronization cannot be maintained. In chaos shift keying (CSK) [7,8], a message signal is encoded by transmitting one chaotic signal for a binary signal ''1'' and another chaotic signal for ''0.'' These two chaotic signals, with similar statistical properties, come from two different systems (or the same system with different systematic parameters). For example, a binary message can be transmitted by switching between two chaotic attractors. At the receiving end, the message signal can be extracted by coherent and noncoherent demodulation technique. The main disadvantage of chaos shift keying is the need for resynchronization when symbols change, giving poor bandwidth efficiency. Finally, in chaotic modulation [9–11], the message signal is injected into a chaotic system as a bifurcation ''parameter,'' with the range of the bifurcation parameter chosen to guarantee motion in a chaotic region. The main advantage of the chaotic modulation scheme is that it does not require any code synchronization, which is necessary in traditional SS communication systems with coherent demodulation techniques. The crucial design factor is, however, the retrieval of the bifurcation parameter variation from the receiving SS signal which may be distorted by channel noise. The main purpose of this paper is to propose and realize an effective approach for chaotic demodulation.

Because of their ability in modeling any arbitrary nonlinear real-valued map defined on compact real sets [12], the radial-basis-function (RBF) neural networks have been employed for the identification of nonlinear dynamical systems [13–16]. In this paper, we specifically describe an on-line adaptive demodulator for a chaotic modulation communication system based on an RBF network. This demodulator can adaptively retrieve message signals from receiving SS signals, which are contaminated by channel noise. Two assumptions are made in our design. Firstly, the transmitter's parameters vary slowly with time, so that the modulation system included at the transmitter can be seen as an autonomous system in a time interval. Secondly, the communication channel is distorted by additive white Gaussian noise (AWGN). The paper is organized as follows. In the next section we give a brief review of chaotic modulation and demodulation. In Sec. III, a Henon-map based demodulation process is formulated. Using an RBF network with an adaptive learning algorithm, a new demodulator for extracting message signals from receiving signals is proposed in Sec. IV. Simulation results are presented in Sec. V for verification. Some remarks regarding practical implementation are included in Sec. VI.

II. REVIEW OF CHAOTIC MODULATION

Recently, a number of schemes for chaotic modulation communication systems have been proposed. Corron and

*Email address: fengjc@swnu.edn.cn
†Email address: cktse@eie.polyu.edu.hk

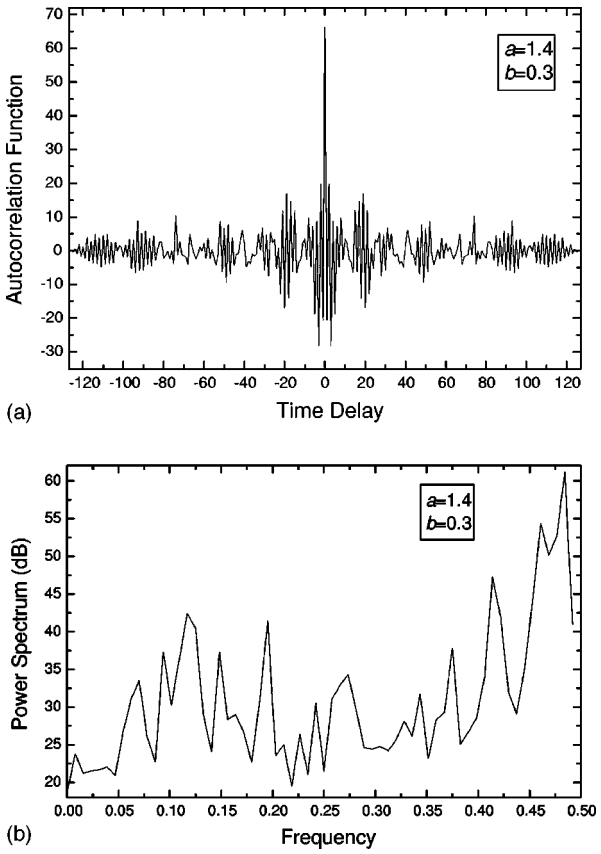

FIG. 1. (a) Autocorrelation function and (b) power spectrum of x_1 for the Henon map with $a=1.4$ and $b=0.3$.

Hahs [17] proposed a drive-response (master-slave) based chaotic modulation communication system for noise-free channels. The demodulation task was realized by a nonlinear filter. Another drive-response based demodulator that can operate under a noise-free channel was also suggested by Sharma and Poonacha [18], which is based on minimization of a cost function to estimate the transmitter's parameters (i.e., messages) by using a gradient search algorithm. Anishchenko and Pavlov [19] proposed a global reconstruction approach for extracting bifurcation parameters of a class of chaotic modulation communication systems. This modulation-demodulation strategy was effective and reliable if, and only if, the motion equation of the transmitter can be rewritten in the following form:

$$\frac{dx_1}{dt}=x_2,\quad \frac{dx_2}{dt}=x_3,\quad \frac{dx_3}{dt}=x_4,\quad \ldots,\quad \frac{dx_E}{dt}=f(X,\Gamma), \tag{1}$$

where $X=[x_1,x_2,\ldots,x_E]^T$ is the vector of equivalent state variables of a chaotic system, and Γ is the parameter vector of the system. When deriving Eq. (1) from the original set of state variables of the transmitter, one usually encounters singularity due to the existence of a zero denominator [19]. Such singularity may lead to an undesired high-frequency component in the retrieved message signal. In Dedieu and Orgorzalek [20], it was demonstrated that low-dimensional chaotic systems can be structurally identifiable via an optimization technique based on minimizing a chosen cost function, e.g., the second-order quasi-Newton algorithm [20]. In the identification process, however, one has to design an effective adaptive algorithms in order to avoid being trapped in local minima of the cost function during the optimal process. In Müller and Elmirghani [21], an artificial-neural-network-based chaotic transmission strategy for the one-dimensional logistic map was proposed. This approach employs a RBF neural network with a fixed hidden unit number to approximate SS signals. However, in reconstructing a chaotic system, there is no systematic method to select a suitable number of hidden layer units, especially for high-dimensional chaotic systems [22,23]. Thus, the design of an appropriate RBF network for a general class of signals can be a complicated task. In the following, this problem is addressed using an on-line adaptive learning algorithm.

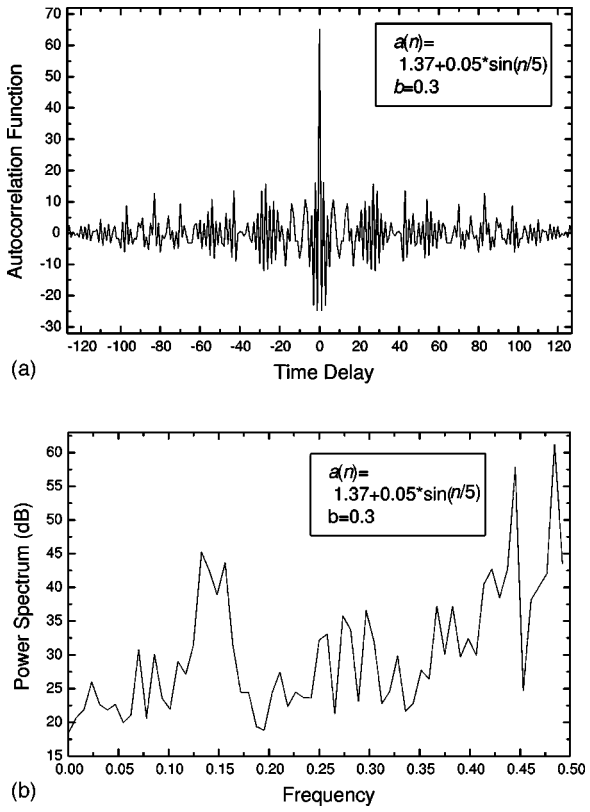

FIG. 2. (a) Autocorrelation function and (b) power spectrum of x_1 for the Henon map with $a(k)=1.37+0.05\sin(k/5)$ and $b=0.3$.

III. FORMULATION OF CHAOTIC MODULATION AND DEMODULATION

Consider the Henon map

$$x_1(k+1)=1-ax_1^2(k)+x_2(k), \tag{2}$$

$$x_2(k+1)=bx_1(k),$$

where a is the bifurcation parameter, and b is fixed at a chosen value. The broadband feature of this map can be verified by inspecting the autocorrelation function and power spectrum of x_1, when a is fixed or time varying. If $a(k)$ is

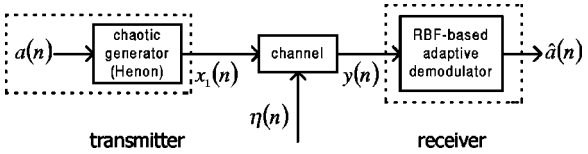

FIG. 3. Block diagram representation of the chaotic modulation communication system.

time varying, we choose its range to ensure chaotic motion of the system. Figure 1 shows the autocorrelation function and power spectrum of $x_1(k)$ when a is fixed at 1.4. Also, Fig. 2 shows the autocorrelation function and power spectrum of $x_1(k)$ when a is varied according to

$$a(k)=1.37+0.05\sin(k/5), \quad \text{for } k=1,2,3,\ldots. \tag{3}$$

We clearly see that x_1 is a highly uncorrelative signal and occupies a broad spectrum, for both cases of fixed and time-varying a. This property is desirable for SS communication, and hence we may choose x_1 as a transmission signal with $a(k)$ being the message signal.

In general, when the SS signal x_1 passes through a practical channel with AWGN, the receiving signal $y(k)$ is

$$y(k)=x_1(k)+\eta(k), \tag{4}$$

where $\eta(k)$ is AWGN. One key function of the receiver is to estimate or track x_1 from the receiving signal $y(k)$. It has been demonstrated [24] that for a low-dimensional noisy chaotic attractor, noise can be removed or reduced by projecting the chaotic attractor onto a higher-dimensional subspace. Such a subspace corresponds to the input space of an RBF network in this paper, and by applying the Takens embedding theorem [25], the dimension of this subspace is as low as 5 for the two-dimensional Henon attractor.

Specifically, in our approach, $x_1(k)$ will be estimated from previously observed data $y(k-1), y(k-2), \ldots, y(k-M)$, where $M=5$. For brevity, we define

$$Z(1)=\begin{bmatrix} y(M) \\ y(M-1) \\ \vdots \\ y(1) \end{bmatrix},$$

$$Z(2)=\begin{bmatrix} y[2(M+1)-1] \\ y[2(M+1)-2] \\ \vdots \\ y(M+2) \end{bmatrix},\cdots,$$

$$Z(n)=\begin{bmatrix} y[n(M+1)-1] \\ y[n(M+1)-2] \\ \vdots \\ y[n(M+1)-M] \end{bmatrix},$$

where $Z(.)$ also stands for $[z_1,z_2,\ldots,z_M]^T$. Note that $Z(n)$ and $y[n(M+1)]$ together form one complete observation.

To avoid confusion, we define an observation step as the duration for one complete observation, i.e., the time for reading $(M+1)$ data points. The problem is effectively reduced to a one-step-ahead prediction, which can be formulated as

$$\hat{x}_1[n(M+1)]=h[Z(n)], \tag{5}$$

where $\hat{x}_1[n(M+1)]$ is the estimate for $x_1[n(M+1)]$ and $h(.)$ is a nonlinear function that can be realized by an RBF neural network with an adaptive learning algorithm, as will be proposed in the next section. After x_1 is tracked, the second equation in Eq. (2) can be used to estimate x_2. We will call $(\hat{x}_1,\hat{x}_2)$ an estimate point pair, which is available every observation step, i.e., $(M+1)$ time steps.

To estimate a, we will make use of the first equation of Eq. (2), which can be rearranged as

$$\hat{a}(k)\hat{x}_1^2(k)-1=\hat{x}_2(k)-\hat{x}_1(k+1). \tag{6}$$

If $a(k)$ is a constant within a window of T_1 observation steps [i.e., $T_1(M+1)$ time steps], then the Henon map can be seen as an autonomous system in the window, and $\hat{a}$ can be estimated by a least-squares-fit approach. Specifically, to find $\hat{a}$, we use the following formula, which requires L samples of $(\hat{x}_1,\hat{x}_2)$, at intervals of T_2 observation steps:

$$\hat{a}=\frac{\sum_{n=1}^{L}[\{\hat{x}_1^2[nT_2(M+1)]-\overline{\hat{x}_1^2}\}\{\hat{x}_2[nT_2(M+1)]-\hat{x}_1[nT_2(M+1)+1]-\overline{\hat{x}_2}+\overline{\hat{x}_1}\}]}{\sum_{n=1}^{L}\{\hat{x}_1^2[nT_2(M+1)]-\overline{\hat{x}_1^2}\}^2}, \tag{7}$$

where $T_1>LT_2$, and $\overline{\hat{x}_1^2}$, $\overline{\hat{x}_2}$, and $\overline{\hat{x}_1}$ are, respectively, the mean of estimated values $\hat{x}_1^2[nT_2(M+1)]$, $\hat{x}_2[nT_2(M+1)]$, and $\hat{x}_1[nT_2(M+1)+1]$, for $n=1,2,\ldots,L$. Thus, in LT_2 observation steps, we will make available one estimate of a which is given by Eq. (7).

A block diagram representation of this chaotic modulation communication system is shown in Fig. 3.

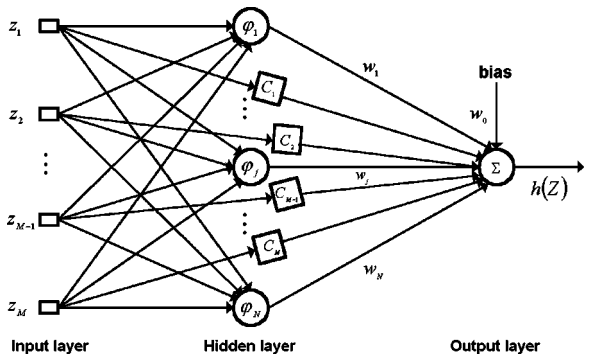

FIG. 4. RBF network configuration.

IV. ON-LINE ADAPTIVE LEARNING ALGORITHM AND DEMODULATION

A. Overview of radial basis function neural network

The RBF network is a three-layer neural network, [13,26] comprising an input layer, a hidden layer, and an output layer, as shown in Fig. 4. The input layer consists of M units, connecting the input vector, for example $Z(n)$ in our application. The ith input unit is directly connected to the output unit through a gain factor c_i, and the ith hidden unit is connected to the output unit through a weight factor w_i. Effectively, the network performs a nonlinear mapping from the input space $\Re^M$ to the output space $\Re$, which is described by

$$h[Z(n)]=w_0+\sum_{i=1}^{M} c_i z_i+\sum_{i=1}^{N} w_i \varphi_i[Z(n)], \tag{8}$$

where w_0 is the bias term. The function $\varphi_i: \Re^M \rightarrow \Re$ is called *activation function* and is given generally by

$$\varphi_i(Z)=\varphi(\|Z-Q_i\|), \tag{9}$$

where $Q_i \in \Re^M$ is known as the RBF center, and $\|\bullet\|$ denotes a distance measurement. The Euclidean distance is adopted in this paper. Moreover, it has been shown [14] that the choice of the nonlinear function φ is not critical to the performance of the RBF network. Typical choices include, for instance, the thin plate spline function $x^2 \log(x)$, the Gaussian function $\exp(-x^2/\sigma^2)$, the multiquadric function $(x^2+\sigma^2)^{1/2}$ and the inverse multiquadric function $(x^2+\sigma^2)^{-1/2}$. All these activation functions have proven good approximation capability regardless of their asymptotic properties [14,26]. In the following, we will use the Gaussian function

$$\varphi_i[Z(n)]=\exp\left(-\frac{\|Z(n)-Q_i(n)\|^2}{2\sigma_i^2}\right), \tag{10}$$

where σ_i is the width of the Gaussian activation function of the ith hidden unit. By putting Eq. (10) in Eq. (8), we have

$$h[Z(n)]=w_0+\sum_{i=1}^{M} c_i(n) z_i+\sum_{i=1}^{N} w_i \times \exp\left(-\frac{\|Z(n)-Q_i(n)\|^2}{2\sigma_i^2}\right). \tag{11}$$

B. Network growth

The network begins with no hidden layer unit. As signal y is received, the network grows by creating new hidden units and connecting the received data to the new hidden units. Precisely, given an observation $\{Z(n), y[n(M+1)]\}$, the criteria for creating a new hidden unit are

$$\|Z(n)-Q_{nr}\|>\eta_1, \tag{12}$$

$$\epsilon(n)=y[n(M+1)]-h[Z(n)]>\eta_2, \tag{13}$$

$$\epsilon_{\mathrm{rms}}^n=\sqrt{\frac{\sum_{i=n-T_3+1}^{n}\{y[i(M+1)]-\hat{x}_1[i(M+1)]\}^2}{T_3}}>\eta_3, \tag{14}$$

where Q_{nr} is the center of the hidden unit, which is nearest $Z(n)$, T_3 is the number of observation steps of a sliding data window covering a number of latest observations for computing the output error, and η_1, η_2 and η_3 are thresholds. Specifically, $\eta_1=\max(\eta_{\max}\beta^n, \eta_{\min})$, where β is a decaying factor, and $\eta_{\max}$ and $\eta_{\min}$ are the maximum and minimum of η_1. The first criterion essentially requires that the input be far away from stored patterns, the second criterion requires that the error signal be significant, and the third criterion specifies that within the sliding data window of T_3 observation steps, the root-mean-square (rms) error is also significant. Now suppose the $(N+1)$th hidden unit is to be added to the network. The parameters associated with this new unit are assigned as follows:

$$w_{N+1}=\epsilon(n), \tag{15}$$

$$Q_{N+1}=Z(n), \tag{16}$$

$$\sigma_{N+1}=\rho\|Z(n)-Q_{nr}\|, \tag{17}$$

where ρ $(\rho<1)$ is an overlap factor that controls the extent of overlap of the responses of the hidden units for an input.

C. Network update with extended Kalman filter

When the observation $\{Z(n), y[n(M+1)]\}$ does not satisfy the criteria (12) to (14), no hidden unit will be added, and the extended Kalman filter (EKF) is then used to adjust the parameters of the network. These parameters define the state vector V of the network,

$$V=[c_1, c_2, \ldots, c_M, w_0, w_1, Q_1^T, \sigma_1, \ldots, w_N, Q_N^T, \sigma_N]^T. \tag{18}$$

Thus, we can write the gradient vector of $h(.)$ with respect to V as

$$B[Z(n)]=\frac{\partial h(.)}{\partial V}=\Bigg[z_1,z_2,\ldots,z_M,1,\phi_1[Z(n)],\phi_1[Z(n)]\times\frac{w_1}{\sigma_1^2}[Z(n)-Q_1]^T,\phi_1[Z(n)]\times\frac{w_1}{\sigma_1^3}\|Z(n)-Q_1\|^2,\ldots,\phi_N[Z(n)],\phi_N[Z(n)]\frac{w_N}{\sigma_N^2}[Z(n)-Q_N]^T,\phi_N[Z(n)]\times\frac{w_N}{\sigma_N^3}\|Z(n)-Q_N\|^2\Bigg]. \quad (19)$$

Now, denote the corrected error covariance matrix of V at instant $(n-1)$ by $P(n-1,n-1)$. Then, the current estimate of the error covariance matrix can be found from the following relation:

$$P(n,n-1)=IP(n-1,n-1)I^T=P(n-1,n-1), \quad (20)$$

where I is an identity matrix. Other parameters used in the EKF algorithm are the variance $R(n)$ of y as defined in Eq. (4) and the Kalman gain vector $K(n)$, whose propagation equations at instant n satisfy with

$$R(n)=B[Z(n)]P(n,n-1)B^T[Z(n)]+R_D. \quad (21)$$

$$K(n)=P(n,n-1)B^T[Z(n)]/R(n), \quad (22)$$

where R_D is the variance of the measured noise. Having computed $K(n)$, we can then update the state vector according to

$$V(n)=V(n-1)+K(n)\epsilon(n), \quad (23)$$

where $V(n)$ and $V(n-1)$ are, respectively, the state vector of the present and previous observation step. Finally, the error covariance matrix is corrected according to

$$P(n,n)=\{I-K(n)B[Z(n)]\}P(n,n-1)+\gamma I, \quad (24)$$

where γ is a small scaling factor introduced to improve the RBF network's adaptability to future input observations in the case of very rapid convergence of the EKF algorithm. [12] Finally, it is worth noting that when a new unit is added to the hidden layer, the dimension of $P(n,n)$ changes, as can be seen from the following relation:

$$P(n,n)=\begin{bmatrix}P(n-1,n-1) & \mathbf{0}_1\\ \mathbf{0}_2 & p_0 I\end{bmatrix}, \quad (25)$$

where $\mathbf{0}_1$ and $\mathbf{0}_2$ are zero matrices of appropriate dimension, and p_0 is a constant representing an estimate of the uncertainty in the initial values assigned to the network parameters, which in this algorithm is also the variance of the observation $\{Z(n),y[n(M+1)]\}$.

D. Pruning of hidden units

As the network grows, the number of hidden units increases, and so will the computing complexity. Moreover, some added hidden units may subsequently end up contributing very little to the network output. The network will only benefit from those hidden units in which the input patterns are close to the stored patterns. Thus, pruning redundant units in the hidden layer becomes imperative. We denote the weighted response of the ith hidden unit for input $Z(n)$ as

$$u_i(n)=w_i\varphi_i, \quad \text{for } i=1,2,\ldots,N. \quad (26)$$

Suppose the largest absolute output value for the nth input $Z(n)$ among all hidden unit's weighted outputs is $|u_{\max}(n)|$. Also denote the normalized output of the ith hidden unit for the nth input as

$$\xi_i(n)=\left|\frac{u_i(n)}{u_{\max}(n)}\right|. \quad (27)$$

In order to keep the size of the network small, we need to remove hidden units when they are found noncontributing. Essentially, for each observation, each normalized output value $\xi_i(n)$ is evaluated. If $\xi_i(n)$ is less than a threshold θ for T_3 consecutive observations, then the ith hidden unit should be removed, thereby keeping the network size and the computing complexity to a minimum.

E. Summary of the flow of algorithm

Basically, the above adaptive algorithm aims to retrieve a value of a for each time window of T_1 observation steps. Two types of estimation are performed by this algorithm. In one observation step [i.e., $(M+1)$ time steps], an estimate of (x_1,x_2) is produced. This reconstructs the SS signal. Then, in LT_2 observation steps, an estimate of a is produced. This retrieves the message. Moreover, the latter estimation requires knowledge of the former, and in practice the RBF network needs a number of observation steps to converge its weights and coefficients. Thus, in our algorithm, we allow a subwindow of T_4 observation steps ($T_4<T_1$), during which estimation of a is omitted. In the remaining subwindow of (T_1-T_4) observation steps, estimation of a is then performed to retrieve the message signal.

Specifically, the purpose in the first subwindow of T_4 observation steps is mainly to train the network to track the dynamics, and in the next subwindow of (T_1-T_4) observation steps, the ''trained'' network estimates x_1 using

$$\hat{x}_1[nT_2(M+1)+1]\doteq h[Z'(n+1)], \quad (28)$$

$$\hat{x}_1[nT_2(M+1)+2]\doteq h[Z''(n+1)], \quad (29)$$

where

$$Z'(n+1)=\begin{bmatrix} \hat{x}_1[nT_2(M+1)] \\ y[nT_2(M+1)-1] \\ \vdots \\ y[nT_2(M+1)-(M-1)] \end{bmatrix},$$

$$Z''(n+1)=\begin{bmatrix} \hat{x}_1[nT_2(M+1)+1] \\ \hat{x}_1[nT_2(M+1)] \\ y[nT_2(M+1)-1] \\ \vdots \\ y[nT_2(M+1)-(M-2)] \end{bmatrix}, \quad (30)$$

and the least-squares-fit is used to retrieve the message a, i.e., using Eq. (7). The following pseudocodes summarize the demodulation algorithm:

initialize the networks,
for each message signal $a(k)$
  for each observation $\{Z(n),y[n(M+1)]\}$ do
  compute networks output subject to Eq. (11),
  determine whether or not a hidden unit should be added,
if conditions (12) to (14) hold
  add a new hidden unit,
  assign relevant parameters,
  adjust covariance matrice (25),
else
  adjust the networks parameters according to Eqs. (18) to (24)
endif;
check the criterion for pruning a hidden unit,
if $\xi_i(n)<\theta$ for T_3 continuous observations
  delete the ith hidden unit,
  reduce the network size,
endif;
if $n \geqslant T_4$
  estimate $x_1[nT_2(M+1)+1]$ and $x_1[nT_2(M+1)+2]$ with the RBF network,
  perform demodulation with Eq. (7),
  endif;
  end for;
end for.

In the next section, we will implement the above demodulation algorithm in an RBF network and apply the network to extract messages from broadband signals.

V. COMPUTER SIMULATIONS AND EVALUATION

Four different kinds of message signals will be employed to test the proposed demodulation scheme, namely, square-wave, sine-wave, speech, and image signals. The square-wave signal is defined by the following piecewise linear function:

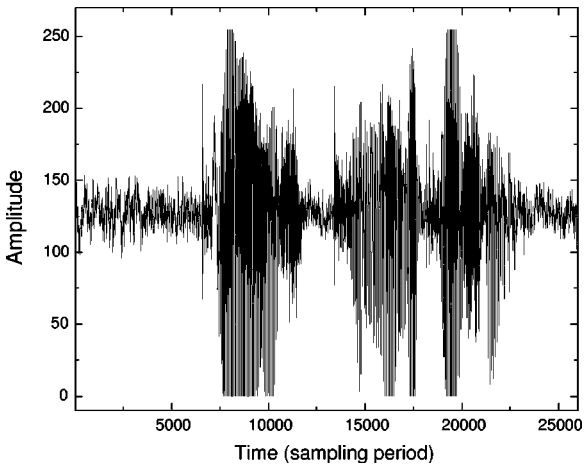

FIG. 5. A male speech wave form: ''CHAOS COMMUNICATIONS,'' sampled at 11 kHz and 8 bit precision.

$$a(k)=\begin{cases} 1.37, & k\in[1,551] \\ 1.42, & k\in[552,1102] \\ 1.35, & k\in[1103,1653] \\ 1.39, & k\in[1654,2204]. \\ 1.32, & k\in[2205,2755] \\ 1.36, & k\in[2756,3306] \\ 1.41, & k\in[3307,3857] \end{cases} \quad (31)$$

The sine-wave signal is as defined in Eq. (3) for $k=1$ to 200. The speech signal used in the test contains a male speech signal ''CHAOS COMMUNICATIONS,'' as shown in Fig. 5, which is sampled at 11 kHz and 8 bit precision. Finally, the image signal is from an Einstein portrait with 192×213 pixels, each pixel having 256 gray levels, as shown in Fig. 6. In this proposed demodulating algorithm, each message signal $a(k)$ is constant in a time window of T_1 observation steps. For example, each pixel value of the image signal is constant in its T_1 observation steps, and each sampling value of the speech signal is also constant in its T_1 observation steps.

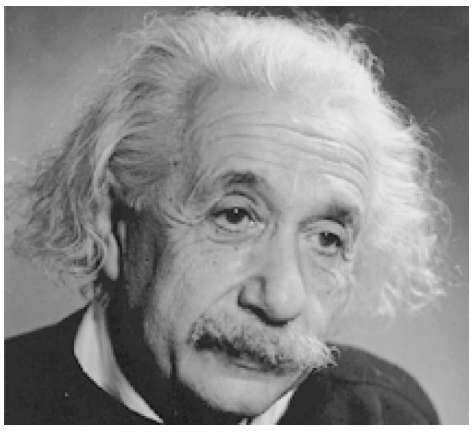

FIG. 6. Einstein portrait (192×213 pixels).

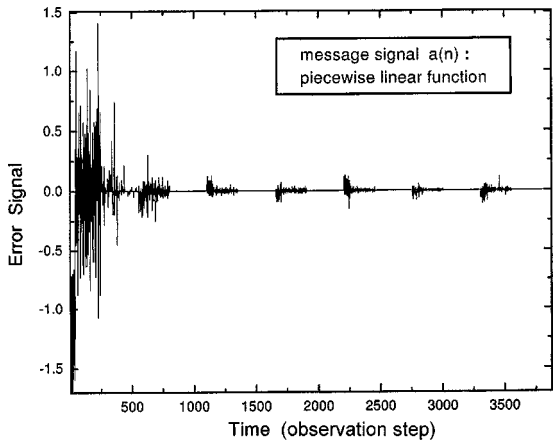

FIG. 7. Error wave form of x_1 for the case of square-wave message.

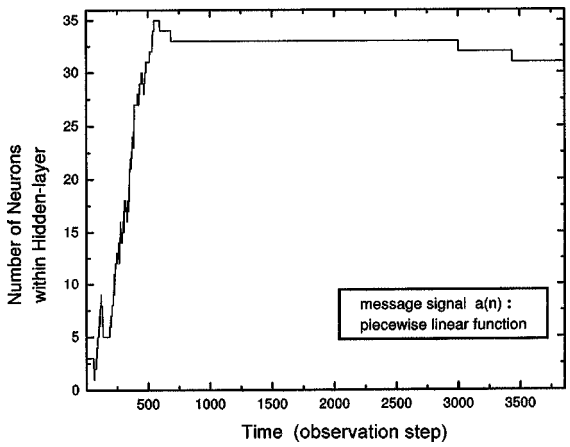

FIG. 8. Number of hidden units showing growth profile for the case of square-wave message.

In the simulation, the transmitted SS signal is controlled to reach the required signal-to-noise ratio (SNR) value in the AWGN channel. For the retrieved signal, the mean-square error (MSE) is used to evaluate the demodulator's performance. The following definition for MSE is adopted.

$$\mathrm{MSE}=10\ \log_{10}\left[\frac{1}{T_5}\sum_{n=1}^{T_5}[\hat{a}(n)-a(n)]^2\right], \qquad (32)$$

where T_5 is the number of the sampled message signals. In our simulation, the parameters of the RBF network and the EKF are assigned as follows: $T_1=551$, $T_2=5$, $T_3=40$, $T_4=250$, $L=60$, $\eta_2=0.05$, $\eta_3=0.07$, $\eta_{\max}=2.0$, $\eta_{\min}=0.02$, $\rho=0.973$, $p_0=15.0$, $\gamma=0.01$, $\beta=0.997$, and $\theta=0.001$.

We will use the square-wave message example to illustrate a few important performance areas, namely, the error propagation, the network growth profile, and the adaptive movement of the hidden units centers.

(1) For the square-wave message and an SNR of 20 dB in the received SS signal, the error wave form of x_1 is shown in Fig. 7, which shows that in the window of the first T_1 observation steps, the error is the largest among all subsequent windows of T_1 observation steps. This is because the tracking of the dynamics is mainly done in the first window. In each subsequent window of T_1 observation steps, it is found that the error signal in the subwindow of the first T_4 observation steps is larger than that in the rest of the window. The subwindow from (T_4+1) to T_1 is, in fact, the estimation window during which the message signal is evaluated according to Eq. (7).

(2) Illustrated in Fig. 8 is the growth of the hidden layer. It can be seen that the variation of the number of hidden units in the first window is most drastic. Specifically, in the window of the first T_1 observation steps, the number of the hidden units adaptively add and drop with time as well as with the input pattern. In the subsequent windows, the number of the hidden units varies with the input pattern adaptively.

(3) Figure 9 shows the variation of the first and second components of the first unpruned hidden unit's center vector. From this, we can again see that the first window experiences the most rapid change.

It should be clear that the RBF-based demodulator can track the time-varying chaotic system by adaptively adjusting both the number and center positions of the hidden layer units. The message signal and retrieved message signal are shown in Fig. 10 and Fig. 11, for the cases of the square-wave and sine-wave test messages, with SNR = 15dB. Also, the retrieved speech signal and Einstein portrait are shown in Fig. 12 and Fig. 13, respectively.

Finally, we measure the MSE performance for the test message signals. Results are shown in Fig. 14, from which we can see that the MSE decreases as the channel SNR increases. At an SNR of 15dB, the MSE of the four retrieved signals are, respectively, −21.1dB, −19.1dB, −22.9dB, and −16.8dB for the square-wave, sine-wave, speech signal, and image signal.

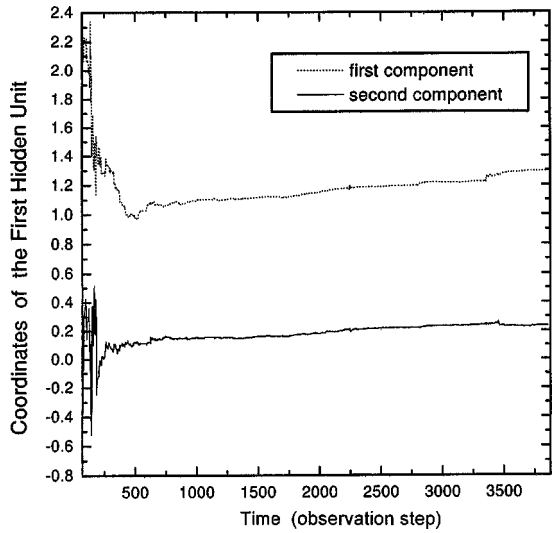

FIG. 9. Variation of the first component (dotted line) and the second component (solid line) of the first unpruned hidden unit's center.

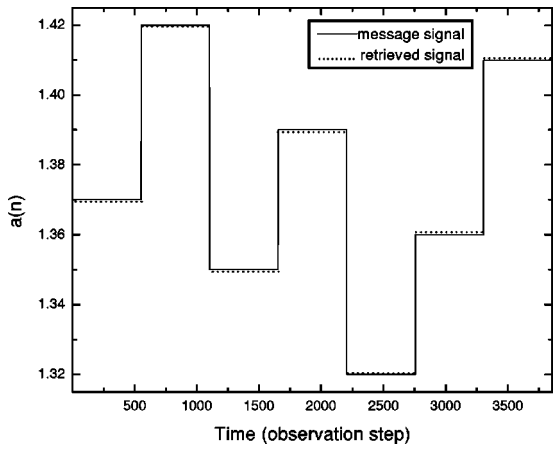

FIG. 10. Square-wave signal (solid line) and retrieved square-wave signal (dotted line, SNR=15dB).

Remarks

It is of interest to note that the estimation of the parameter $a(k)$ can be performed using direct computation based on Eq. (2), similar to the method described in Anishchenko and Pavlov [19]. However, direct computation achieves reasonable performance only when the noise level is very low. Our method shows significant improvement over direct computation in the presence of noise. As a comparison, we present here the MSE's corresponding to the above same examples using the method of direct computation as in Anishchenko and Pavlov [19]. In brief, the method of direct computation [19] involves using the noisy data to solve a from Eq. (2) at each observation step. Then, the average value of a over T_1 observation steps gives an estimate for the message signal. From Fig. 15, we note that when the SNR of the received signal is 15dB, the MSE of the four retrieved signals are 69.5dB, 89.9dB, 139.2dB and 145.8dB for the square-wave, sine-wave, speech signal and image signal, respectively. Thus, we can see the significant improvement that can be gained from using the proposed method, especially for applications in noisy environment. In fact, at a high noise level (low SNR), the chaotic dynamics of Eq. (2) will be drastically altered, invalidating the conventional demodulation approaches for parameter estimation [28]. It should be noted that since our interest is to deal with noisy signals, the variance of the added random noise used in our paper ($\approx 10^{-1}$) is much higher than that used in Anishchenko and Pavlov [19] ($\approx 10^{-5}$).

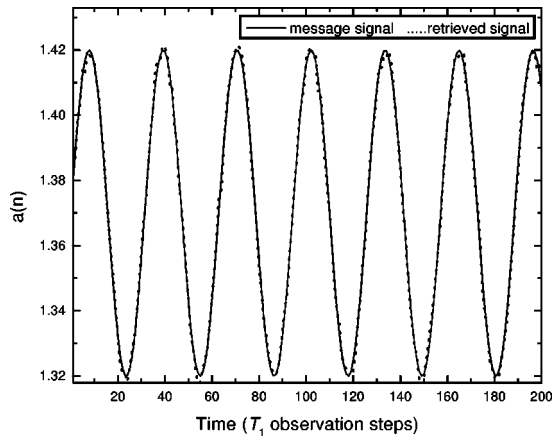

FIG. 11. Sine-wave signal (solid line) and retrieved sine-wave signal (dotted line, SNR=15dB).

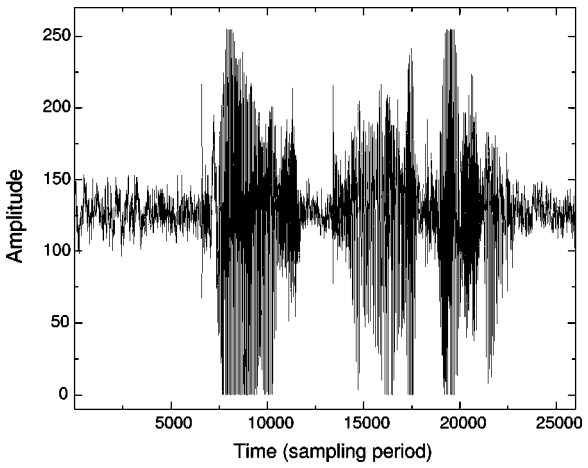

FIG. 12. Retrieved speech signal (MSE=−22.9dB, SNR =15dB).

VI. DISCUSSION

In realizing chaotic communication systems, two areas of performance are important, namely, robustness to external perturbations (noise and parameter mismatches) and security, which are known to be a paradox in communication with chaos [27].

The first one concerns with the system immunity against parameter mismatches and noise in the channel. In our proposed system, variation of the system parameter is exploited for transmitting messages, and resynchronization at the receiver (which is sensitive to external perturnbations) is not required. Hence, parameter mismatches do not present problems to our system, as they may do in other chaotic communication systems.

The second area of performance concerns the security of the system. In the chaotic modulation system under study, the receiver is assumed to ''know'' the transmitter. This of-

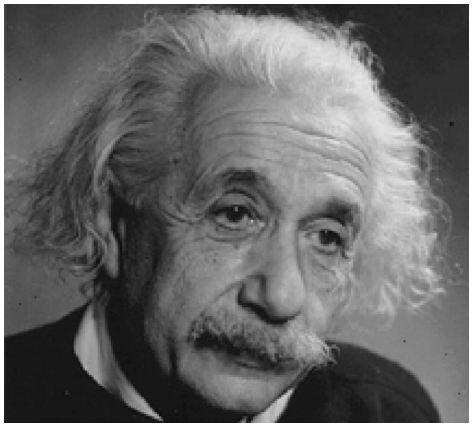

FIG. 13. Retrieved Einstein portrait (SNR=15dB).

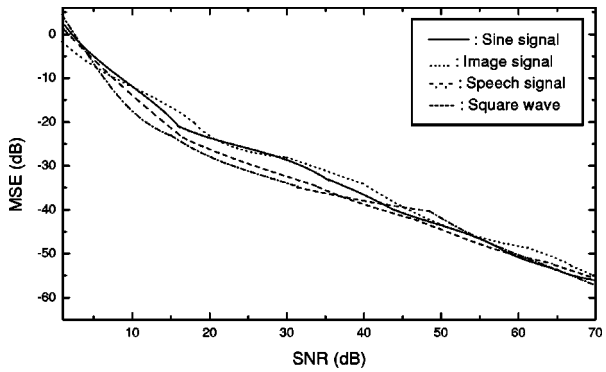

FIG. 14. MSE of retrieved source signals versus SNR using the proposed method.

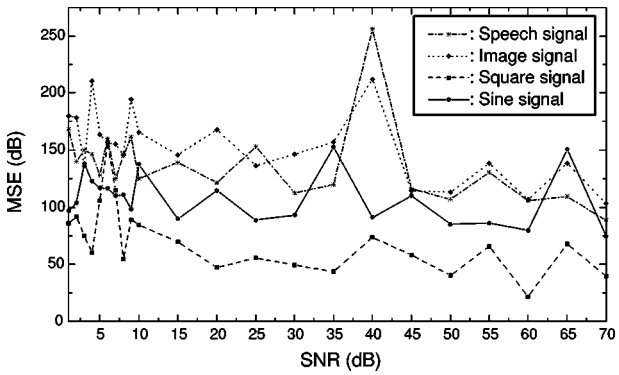

FIG. 15. MSE of retrieved source signals versus SNR using direct computation (for comparison) (Ref. [19]).

fers the basic level of security because, without knowledge of the kind of functions and parameter variation involved in the transmitter, it is almost impossible for an intruder to recover the transmitted message. When the transmitter is ''known,'' moreover, security becomes limited since an intruder may be able to perform tracking and parameter estimation through a different means, such as using the adaptive-control based method studied by Zhou and Lai [27]. It should be noted that tracking in a highly noisy environment as studied in this paper is still a very difficult task, making it nontrivial for an intruder to recover the message even with good knowledge of the transmitter [22,27]. Furthermore, one may also achieve a greater level of security if some cryptographic feature is incorporated in the modulation process. This, however, is beyond the scope of the present paper.

VII. CONCLUSION

In this paper, we have designed an on-line adaptive demodulator for recovering message signals that are transmitted through chaotic carriers contaminated by additive white Gaussian noise. The message is injected into a Henon-map-based transmitter system as variation of a bifurcation parameter. The demodulation process essentially aims to recover the embedded parameter variation. The essential component of the proposed demodulator is a radial-basis-function neural network, which incorporates an adaptive learning algorithm to track the dynamics of the Henon map. The least-squares fit is used to estimate message signals. The purpose of the adaptive learning algorithm is to adaptively configure hidden layer size and adjust the relevant parameters with an extended Kalman filter algorithm. The system is tested with square-wave, sine-wave, image, and speech signals serving as messages. Results have demonstrated that the demodulator is capable of performing the required demodulation task for a noisy communication channel.

ACKNOWLEDGMENTS

This work was supported in part by Hong Kong University Research Grants Council under a competitive-bid earmarked grant, and by Hong Kong Polytechnic University Research Committee.

[1] T. Yamada and H. Fujisaka, Prog. Theor. Phys. **69**, 32 (1983).
[2] T. Yamada and H. Fujisaka, Prog. Theor. Phys. **70**, 1240 (1983).
[3] L. M. Pecora and T. L. Carroll, Phys. Rev. Lett. **64**, 821 (1990).
[4] L. M. Pecora and T. L. Carroll, Phys. Rev. A **44**, 2374 (1990).
[5] K. Kocarev, K. S. Halle, K. Eckert, L. O. Chua, and U. Parlitz, Int. J. Bifurcation Chaos Appl. Sci. Eng. **2**, 709 (1992).
[6] K. M. Cuomo and A. V. Oppenheim, Phys. Rev. Lett. **71**, 65 (1993).
[7] H. Dedieu, M. P. Kennedy, and M. Hasler, IEEE Trans. Circuits Syst., II: Analog Digital Signal Process. **40**, 634 (1993).
[8] U. Parlitz, L. O. Chua, L. Kocarev, K. S. Halle, and A. Shang, Int. J. Bifurcation Chaos Appl. Sci. Eng. **2**, 973 (1996).
[9] K. S. Halle, C. W. Wu, M. Itoh, and L. O. Chua, Int. J. Bifurcation Chaos Appl. Sci. Eng. **3**, 469 (1993).
[10] L. Kocarev and U. Parlitz, Phys. Rev. Lett. **74**, 5028 (1995).
[11] U. Parlitz, L. Kocarev, T. Stojanovski, and H. Preckel, Phys. Rev. E **53**, 4351 (1996).
[12] V. Kadirkamanathan and M. Niranjan, Neural Comput. **5**, 954 (1993).
[13] C. L. Chen, W. C. Chen, and F. Y. Chang, IEE Proc.-D: Control Theory Appl. **140**, 442 (1993).
[14] P. R. Chang and W. H. Yang, IEEE Trans. Vehicular Tech. **46**, 155 (1997).
[15] L. Yingwei, N. Sundararajan, and P. Saratchandran, IEE Proc.-D Control Theory Appl. **144**, 202 (1997).
[16] F. Girosi and T. Poggio, Biol. Cybern. **63**, 169 (1990).
[17] N. J. Corron and D. W. Hahs, IEEE Trans. Circuits Syst., I: Fundam. Theory Appl. **44**, 373 (1997).
[18] N. Sharma and P. G. Poonacha, Phys. Rev. E **56**, 1242 (1997).
[19] V. S. Anishchenko and A. N. Pavlov, Phys. Rev. E **57**, 2455 (1998).
[20] H. Dedieu and M. J. Ogorzalek, IEEE Trans. Circuits Syst., I: Fundam. Theory Appl. **44**, 948 (1997).
[21] A. Müller and J. M. H. Elmirghani, IEEE Comm. Lett. **2**, 241 (1998).

[22] S. Papadimitriou, A. Bezerianos, and T. Bountis, Int. J. Bifurcation Chaos Appl. Sci. Eng. **9**, 221 (1999).
[23] D. M. Walker and A. I. Mees, Int. J. Bifurcation Chaos Appl. Sci. Eng. **8**, 557 (1998).
[24] J. C. Principe and J. M. Kuo, Proc. SPIE **2038**, 326 (1993).
[25] F. Takens, Lecture Notes in Mathematics Vol. 898 (Springer-Verlag, Berlin, 1981).
[26] S. Haykin, *Neural Networks: a Comprehensive Foundation* (Prentice-Hall, Englewood Cliffs, NJ, 1994).
[27] C. Zhou and C. H. Lai, Phys. Rev. E **59**, 6629 (1999).
[28] G. A. Johnson, D. J. Mar, T. L. Carroll, and L. M. Pecora, *Proceedings of the Fourth Experimental Chaotic Conference* (World Scientific, Singapore, 1998), p. 407.