Observer-based Tracking and Identification of Chaotic Systems with Application to Chaotic Communication Through Noisy Channel

Jiu-chao Feng and Chi K. Tse
Department of Electronic and Information Engineering, Hong Kong Polytechnic University, Hong Kong, China
Email: fengjc@eie.polyu.edu.hk & eunktse@polyu.edu.hk

Abstract — A novel observer for tracking and identifying a chaotic system with time-varying bifurcation parameters via an observation signal contaminated by additive white Gaussian noise (AWGN) is developed in this paper. It is realized by an adaptive algorithm which takes advantage of the good approximation capability of the Radial Basis Function (RBF) neural network and the ability of the Extended Kalman Filter (EKF) for tracking a time-varying dynamical system. It is demonstrated that, provided the bifurcation parameter varies slowly in a time window, a chaotic dynamical system can be tracked and identified continuously, and the time-varying bifurcation parameter can also be retrieved in a sub-window of time via a simple least-square-fit method. The proposed tracking and parameter retrieval method can be used in a chaotic modulation/demodulation system where a message is represented by the parameter variation and carried by a wide-band chaotic signal.

I. INTRODUCTION

The identification of a chaotic system can be difficult when the observation signal is contaminated by noise and when some parameters are varying with time [1]. Since real systems do have parameter variations and noise always exists in any physical channel, it is of interest to consider the problem of tracking a chaotic signal with time-varying parameters from a noisy observation signal. In this paper, we propose a method for solving the above-mentioned tracking problem using a Radial Basis Function (RBF) neural network assisted by an Extended Kalman Filter (EKF). The result is a novel observer that adaptively reconstructs a chaotic signal. We will also show how the time-varying parameter can be retrieved with a simple least-square-fit. Our combined tracking and parameter retrieval method finds application in communication where a message is represented by the parameter variation and carried by a chaotic signal, as in a typical chaotic modulation system.

II. THE PROBLEM AND OBJECTIVE

We consider the problem of tracking the Henon map which can be represented by the following 2-dim iterative map:

\[
\begin{align*}
x_1(k+1) &= 1 - ax_1^2(k) + x_2(k) \\
x_2(k+1) &= bx_1(k)
\end{align*}
\]

(1)

where \(a\) is a time-varying bifurcation parameter whose variation ensures chaotic motion of the system, and \(b\) is fixed at 0.3. The observed signal is \(y(k)\), which is essentially \(x_1(k)\) with noise added, i.e.,

\[
y(k) = x_1(k) + \eta(k)
\]

(2)

where \(\eta(k)\) is the additive white Gaussian noise (AWGN). Our objective is to track and identify \(x_1\) from \(y(k)\). Because of its ability in modeling any arbitrary nonlinear real-valued map defined on compact real sets [2, 3], an RBF neural network is employed to perform the tracking, which will be assisted by an EKF for coping with the effect of the time-varying parameter.

III. ADAPTIVE TRACKING ALGORITHM

A. Review of Radial-Basis-Function Neural Network

The RBF neural network is a three-layer neural network, comprising an input layer, a hidden layer and an output layer, as shown in Fig. 1. The input layer consists of \(M\) units, connecting the input vector \(Z(n)\), which is constructed from the observation \(y(k)\) [4]. For brevity we write \(Z(.)\) as \([z_1, z_2, \ldots, z_M]^T\), which is defined as

\[
Z(n) = [y(n(M+1)-1) \ y(n(M+1)-2) \ \cdots \ y(n(M+1)-M)]^T
\]

Thus, we have, for example,

\[
Z(1) = [y(M) \ y(M-1) \ \cdots \ y(1)]^T,
\]

\[
Z(2) = [y(2(M+1)-1) \ y(2(M+1)-2) \ \cdots \ y(2+M)]^T,
\]

etc.

The \(i\)th input unit is directly connected to the output unit through a gain factor \(c_i\), and the \(i\)th 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 \(\mathbb{R}^M\) to the output space \(\mathbb{R}\), which is described by

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

(3)

where \(w_0\) is the bias term. The function \(\varphi_i : \mathbb{R}^M \rightarrow \mathbb{R}\) is called activation function and is given generally by

\[
\varphi_i(Z) = \varphi(||Z - Q_i||)
\]

(4)

where \(Q_i \in \mathbb{R}^M\) is known as the RBF center, and \(||\cdot||\) represents Euclidean distance norm. \(\varphi_i\) is selected as the Gaussian function defined by

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

(5)

where \(\sigma_i\) is the width of the Gaussian function of the \(i\)th hidden unit.

One complete observation consists of \(Z(n)\) and \(y(n(M+1))\). For brevity we define \((Z(n), y(n(M+1)))\) as an observation.
pair, and the duration for one complete observation as an observation step, i.e., the time for reading \((M+1)\) data points. The problem is now reduced to a one-step-ahead prediction which can be formulated as

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

(6)

where \(\hat{x}_1(n(M+1))\) is the estimate for \(x_1(n(M+1))\). In this paper, we assume that the bifurcation parameter \(a\) at a time window of \(T_1\) observation steps is constant such that the system can be seen as an autonomous system in the window.

B. Network Growth

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

\[
||Z(n) - Q_{\text{cen}}|| > \eta_1
\]  

(7)

\[
e(n) = y(n(M+1)) - h(Z(n)) > \eta_2
\]  

(8)

\[
e_{\text{rms}}^n = \sqrt{\frac{\sum_{t=n-T_2+1}^{n} [y_t(i(M+1)) - \hat{x}_1(i(M+1))]^2}{T_2}} > \eta_3
\]  

(9)

where \(Q_{\text{cen}}\) is the center of the hidden unit which is nearest \(Z(n)\), \(T_2\) is the number of observation steps of a sliding data window covering a number of latest observations for computing the output error, and \(\eta_1, \eta_2, \eta_3\) are thresholds. Specifically, \(\eta_1 = \max(\eta_{\text{init}}, \beta\eta_{\text{min}})\), where \(\beta\) is a decaying factor, \(\eta_{\text{init}}\) and \(\eta_{\text{min}}\) are the maximum and minimum of \(\eta_1\), \(n\)'s unit is operation step.

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 \(T_2\) 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:

\[
w_{N+1} = e(n)
\]  

(10)

\[
Q_{N+1} = Z(n)
\]  

(11)

\[
\sigma_{N+1} = \rho||Z(n) - Q_{\text{cen}}||
\]  

(12)

where \(\rho (\rho < 1)\) is an overlap factor which 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 pair \([Z(n), y(n(M+1))]\) does not satisfy the criteria (7) to (9), no hidden unit will be added, and the EKF algorithm [5] is then used to adjust the parameters of the network. These parameters define the state vector, \(V\), of the network,

\[
V = [c_1, c_2, \ldots, c_M, w_0, w_1, Q_1^r, \sigma_1, \ldots, w_N, Q_N^r, \sigma_N]^T.
\]  

(13)

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

\[
B(Z(n)) = \frac{\partial h(\cdot)}{\partial V} x(n)
\]  

(14)

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:

\[
P(n, n-1) = IP(n-1, n-1)I^T = P(n-1, n-1),
\]  

(15)

where \(I\) is an identity matrix. Other parameters used in the EKF algorithm are the variance \(R(n)\) of \(y\) as defined in (2) and the Kalman gain vector \(K(n)\), whose propagation equations at instant \(n\) satisfy with

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

(16)

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

(17)

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

\[
V(n) = V(n-1) + K(n)e(n),
\]  

(18)

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

\[
P(n, n) = (I - K(n)B(Z(n)))P(n, n-1) + \gamma I,
\]  

(19)

where \(\gamma\) 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 [6]. 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.

\[
P(n, n) = \begin{bmatrix}
P(n-1, n-1) & 0_1 \\
0_2 & \rho_0 I
\end{bmatrix}
\]  

(20)

where \(0_1\) and \(0_2\) are zero matrices of appropriate dimension, and \(\rho_0\) 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 \(i\)th hidden unit for input \(Z(n)\) as

\[
u_i(n) = w_i \varphi_i, \quad \text{for } i = 1, 2, \ldots, N
\]  

(21)
Suppose the largest absolute output value for the $n$th input $Z(n)$ among all hidden units’ weighted outputs is $|u_{oab}(n)|$. Also denote the normalized output of the $i$th hidden unit for the $n$th input as

$$
\xi_i(n) = \left[ \frac{u_i(n)}{|u_{oab}(n)|} \right].
$$

In order to keep the size of the network small, we need to remove hidden units when they are found non-contributing. Essentially, for each observation, each normalized output value $\xi_i(n)$ is evaluated. If $\xi_i(n)$ is less than a threshold $\theta$ for $T_2$ consecutive observation pairs, then the $i$th hidden unit should be removed, thereby keeping the network size and the computing complexity to minimal. In the next section, we will test the above algorithm.

E. Parameter Estimation

To estimate $a$, we need to know the type of system. For the Henon map of (1), we have

$$
\hat{a}(k) = \hat{x}_2(k) - 1 = \hat{x}_3(k) - \hat{x}_1(k + 1)
$$

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-square-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 $T_3$ observation steps.

$$
\hat{a} = \left\{ \sum_{n=1}^{L} \left[ \left( \hat{x}_2(nT_3(M + 1)) - \overline{\hat{x}_2} \right)(\hat{x}_2(nT_3(M + 1)) - \overline{\hat{x}_2}) \right] \right\}
$$

where $T_1 > LT_3$, and $\overline{\hat{x}_2}$, $\overline{\hat{x}_3}$ and $\overline{\hat{x}_1}$ are respectively the mean of estimated values $\hat{x}_2(nT_3(M + 1))$, $\hat{x}_3(nT_3(M + 1))$ and $\hat{x}_1(nT_3(M + 1) + 1)$, for $n = 1, 2, \cdots, L$. Thus, in $LT_3$ observation steps, we will make available one estimate of $a$ which is given by (24).

IV. Simulation Results

Two different types of signals will be employed as the time-varying parameter to test the proposed algorithm. First we consider a square-wave signal defined by the following piecewise linear function:

$$
a(k) = \begin{cases} 
1.37, & k \in [1, 551] \\
1.42, & k \in [552, 1102] \\
1.35, & k \in [1103, 1653] \\
1.39, & k \in [1654, 2204] \\
1.32, & k \in [2205, 2755] \\
1.36, & k \in [2756, 3306] \\
1.41, & k \in [3307, 3857]
\end{cases}
$$
Then, we consider an image signal representing an Einstein portrait with 192 x 213 pixels, each pixel having 256 grey levels, as shown in Fig. 2. In this proposed algorithm, $a(k)$ 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.

In the simulation, the threshold parameters of the RBF network and the EKF are assigned as follows: $T_1 = 551$, $T_2 = 40$, $T_3 = 5$, $L = 60$, $M = 5$, $\eta_2 = 0.05$, $\eta_3 = 0.07$, $\eta_{	ext{max}} = 2.0$, $\eta_{	ext{min}} = 0.02$, $\rho = 0.973$, $p_0 = 15.0$, $\gamma = 0.01$, $\beta = 0.997$, $\theta = 0.001$.

We will use the time-varying square-wave signal 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 signal to noise ratio (SNR) of 20dB in $y$, the error waveform of $x_1$ is shown in Fig. 3 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 sub-window of the first $T_4$ observation steps ($T_4 = 250$) is larger than that in the rest of the window. The sub-window from ($T_4 + 1$) to $T_1$ is, in fact, the estimation window during which the bifurcation parameter is evaluated by the least-square-fit algorithm.

2. Illustrated in Fig. 4 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 adds and drops 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. Fig. 5 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 proposed observer can track the time-varying chaotic system by adaptively adjusting both the number and center positions of the hidden layer units. When SNR of $y(k)$ is 15dB, the retrieved signals (bifurcation parameters) are shown in Figs. 6 and 7, for the cases of the square-wave and the image signal.

Finally, we measure the Mean Square Error (MSE) performance for the two retrieved signals. Results are shown in Fig. 8. At an SNR of 15dB, the MSE of the two retrieved signals are respectively $-21.1$dB and $-16.8$dB for the square wave and image signal.

V. CONCLUSION

In this paper, we have proposed an adaptive observer for tracking and identifying a chaotic system with time-varying bifurcation parameter from a noisy observation signal. The essential component of the proposed observer is an RBF neural network which is assisted by an EKF algorithm.

REFERENCES