A Chaos Tracker Applied to Non-coherent Detection in Chaos-based Digital Communication Systems

Jiu-chao Feng, Chi K. Tse and Francis C.M.Lau

Department of Electronic and Information Engineering, Hong Kong Polytechnic University, Hong Kong, China

Email: fengjc@eie.polyu.edu.hk, encktse@polyu.edu.hk & encmlau@polyu.edu.hk

Abstract — In chaos-based communication systems, parameter variation in the chaos generator and additive channel noise represent two practical problems that are hard to solve. For chaos-based digital communication, non-coherent detection has the advantage that the receiver does not need to reproduce the same chaos basis function that has been generated in the transmitter. Such reproduction typically requires a fragile operation of chaos synchronisation between the transmitter and the receiver. In this paper, we consider non-coherent detection under the practical condition of the transmitted signal being contaminated by noise and its generating function being subject to strong parameter variation. A novel tracker is proposed for reconstructing the transmitted chaotic signal. This tracker uses a modified radial-basis-function (RBF) neural network which incorporates a learning algorithm for tracking the noisy chaotic signal under parameter variation. Using this tracker, a non-coherent detector is designed for demodulating chaos-shift-keying (CSK) signals in a CSK digital communication system. Computer simulations, in which the Chua's circuit is used as the chaos generator, are presented to demonstrate the tracking ability and CSK demodulation of the proposed method.

I Introduction

Coherent detection and non-coherent detection represent two distinct approaches for detecting chaos-based digital modulated signals such as in chaos-shift-keying (CSK) or differential chaos-shift-keying (DCSK) communication systems. While coherent detection requires the receiver to reproduce the same chaos basis function as in the transmitter, for example by chaos synchronisation [1], non-coherent detection makes use of some distinguishable property (e.g., bit energy) of the chaotic segments representing the different digital symbols [2]. However, additive channel noise and parameter variation in the chaos generator in the transmitter can seriously impair detection at the receiver. In this paper we consider non-coherent detection under the practical condition of the transmitted signal being contaminated by noise and its generating function being subject to strong parameter variation.

The above problem is equivalent to the basic problem of tracking a chaotic system with time-varying parameters from an observation signal that has been contaminated by additive white Gaussian noise (AWGN). In particular, we assume that the parameter is varying at a rate much slower than the motion of the system itself. Our main contribution in this paper is a novel tracker which can be used to reconstruct the transmitted chaotic signal in a chaos-based communication system. The tracking is accomplished in the sense of Taken's reconstruction theory [3]. Specifically, our proposed tracker uses a modified radial-basis-function (RBF) neural network which incorporates a learning algorithm for accomplishing the afore-described tracking task. Finally, using this tracker, a non-coherent detector can be designed for demodulating chaos-shift-keying (CSK) signals in a CSK digital communication system. We will demonstrate the demodulator's performance by computer simulations.

This work is supported in part by a competitive earmarked grant funded by Hong Kong Research Grant Council.

II Reconstruction of Attractor

Consider the chaos generating system

$$\dot{x} = g(x, t) \tag{1}$$

where $x = [x_1,\ x_2,\ \cdots, x_D]^T$ is the D-dimensional state vector of the system, T denotes transposition of a vector or matrix, and g is a smooth nonlinear function defined on $\Re^D \times \Re$. The transmitted signal s is generally given by

$$s = \phi(x(\mathrm{t})) \tag{2}$$

where $\phi(.)$ is a continuous scalar real-valued function. At the receiver, the received (observed) signal is

$$y = s + \eta \tag{3}$$

where η is the channel noise (AWGN). Furthermore, we assume in this paper that the parameters of system (1) are time-varying in a range that keeps the system in the chaotic region. The rate of variation of the parameters is much slower than the chaotic transmitted signal. Thus, in a certain time window, the parameters of the transmitter's chaos generator are constant, allowing system (1) to be viewed as an autonomous system. The essential task of the tracker is, within the time window, to track the transmitted signal s through the impaired signal y.

In the Euclidean space $\Re^M$ the Takens' reconstruction theorem [3] simply says that given s in (2), the chaotic attractor of (1) can be reconstructed from $\tilde{s} =$

0-7803-6685-9/01/$10.00©2001 IEEE

$[s,\ \dot{s},\ \ddot{s},\ \cdots,\ s^{(M-1)}]^T$ where $M \geq (2D+1)$, and $\dot{s}$, $\ddot{s}$, $\cdots s^{(M-1)}$ denote $\frac{ds}{dt}$, $\frac{d^2s}{dt^2}$ $\cdots$ $\frac{d^{M-1}s}{dt^{M-1}}$, respectively. In other words, there exist a function Ψ such that

$$\boldsymbol{x} = \Psi(\tilde{s}, t) \tag{4}$$

where $\Psi = [\psi_1,\ \psi_2,\ \cdots,\ \psi_D]^T$. It should be noted that $s = \phi(\boldsymbol{x}(t)) \stackrel{\text{def}}{=} f_1(\boldsymbol{x}, t)$, $\dot{s} = \nabla_x \phi(\boldsymbol{x}(t)) \cdot \boldsymbol{g}(\boldsymbol{x}, t) \stackrel{\text{def}}{=} f_2(\boldsymbol{x}, t)$ and $s^{(i)} = \frac{ds^{(i-1)}}{dt} = \nabla_x f_i(\boldsymbol{x}, t) \cdot \boldsymbol{g}(\boldsymbol{x}, t) + \frac{\partial f_i(\boldsymbol{x}, t)}{\partial t} \stackrel{\text{def}}{=} f_{i+1}(\boldsymbol{x}, t)$, where $\nabla_x f_i(\boldsymbol{x}, t)$ is the gradient of $f_i(\boldsymbol{x}, t)$ with respective to $\boldsymbol{x}$, and "$\cdot$" denotes the vector dot (inner) product. Also,

$$\tilde{s} = \boldsymbol{f}(\boldsymbol{x}, t) \tag{5}$$

where $\boldsymbol{f} = [f_1,\ f_2,\ \cdots,\ f_M]^T$, and f_i $(i = 1, 2, \cdots, M)$ is a smooth function. Combining (4) and (5), we have

$$\boldsymbol{x} = \Psi(\boldsymbol{f}(\boldsymbol{x}, t), t). \tag{6}$$

Thus, it is generally possible to reconstruct the chaotic attractor of (1) on the higher dimensional space if $\boldsymbol{f}(\cdot)$ and Ψ are known. However, the receiver usually has no exact knowledge of $\boldsymbol{f}$ and Ψ in practical applications even in the absence of noise. Thus, the crucial step is to find $\boldsymbol{f}$ and Ψ in order to realize the reconstruction task. This problem is tackled in this paper by a modified RBF neural network which will be described in the next section.

III RBF Neural Networks and Tracker

The modified RBF network shown in Fig. 1 is a three-layer neural network, comprising an input layer, a hidden layer and an output layer. The input layer consists of M units, connecting the input vector. 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 weigh factor w_i. The input vector that defines the way embedding is done is given by $\boldsymbol{z}(n) = [z_1\ z_2\ \cdots\ z_M]^T$ which is an M-dimensional input vector given by

$$\boldsymbol{z}(n) = [y(n(M+1)-1)\ y(n(M+1)-2)\ \cdots y(n(M+1)-M)]^T. \tag{7}$$

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 network operation can be described by

$$h(\boldsymbol{z}(n)) = w_0 + \sum_{i=1}^{M} c_i z_i + \sum_{i=1}^{N} w_i \varphi_i(\boldsymbol{z}(n)) \tag{8}$$

where w_0 is the bias term of the output unit. Also, φ_i is the Gaussian activation function defined as

$$\varphi_i(\boldsymbol{z}(n)) = \exp\left(-\frac{\|\boldsymbol{z}(n) - Q_i(n)\|^2}{2\sigma_i^2}\right) \tag{9}$$

where Q_i and σ_i are the center and width of φ_i, respectively.

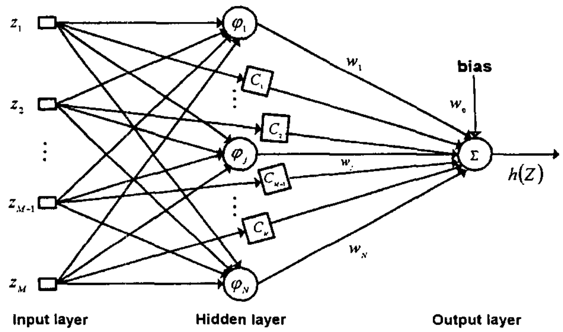

Fig. 1: A modified RBF network configuration

IV Tracking Algorithm

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

$$\|\boldsymbol{z}(n) - Q_{nr}\| > \eta_1 \tag{10}$$

$$\epsilon(n) = y(n(M+1)) - h(\boldsymbol{z}(n)) > \eta_2 \tag{11}$$

$$\epsilon_{\text{rms}}^n = \sqrt{\frac{\sum_{i=n-T_3+1}^{n} \epsilon^2(i)}{T_3}} > \eta_3 \tag{12}$$

where Q_{nr} is the center of the hidden unit which is nearest $\boldsymbol{z}(n)$, 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:

$$w_{N+1} = \epsilon(n) \tag{13}$$

$$Q_{N+1} = \boldsymbol{z}(n) \tag{14}$$

$$\sigma_{N+1} = \rho\|\boldsymbol{z}(n) - Q_{nr}\| \tag{15}$$

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

When the observation $[\boldsymbol{z}(n), y(n(M+1))]$ does not satisfy the criteria (10) to (12), 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, $\boldsymbol{v}$, of the network,

$$\boldsymbol{v} = [c_1, c_2, \cdots, c_M, w_0, w_1, Q_1^T, \sigma_1, \cdots, w_N, Q_N^T, \sigma_N]^T. \tag{16}$$

Now, denote the corrected error covariance matrix of $\boldsymbol{v}$ at time 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:

$$P(n,n-1) = IP(n-1,n-1)I^T = P(n-1,n-1), \quad (17)$$

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

$$R(n) = B(\boldsymbol{z}(n))P(n,n-1)B^T(\boldsymbol{z}(n)) + R_d \quad (18)$$

$$K(n) = P(n,n-1)B^T(\boldsymbol{z}(n))/R(n)\ , \quad (19)$$

where R_d is the variance of the measured noise, and B is the gradient vector of $h(\cdot)$ with respect to $\boldsymbol{v}$. Having computed $K(n)$, we can then update the state vector according to

$$\boldsymbol{v}(n) = \boldsymbol{v}(n-1) + K(n)\epsilon(n), \quad (20)$$

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

$$P(n,n) = (I - K(n)B(\boldsymbol{z}(n)))P(n,n-1) + \gamma I\ , \quad (21)$$

where γ is a small scaling factor introduced to improve the RBF network's adaptability to future observations [4]. 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.

$$P(n,n) = \begin{bmatrix} P(n-1,n-1) & \mathbf{0}_1 \\ \mathbf{0}_2 & p_0 I \end{bmatrix} \quad (22)$$

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 $[\boldsymbol{z}(n), y(n(M+1))]$.

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. Thus, pruning redundant units in the hidden layer becomes imperative. We denote the weighted response of the ith hidden unit for input $\boldsymbol{z}(n)$ as

$$u_i(n) = w_i\varphi_i, \quad \text{for } i = 1, 2, \cdots, N\ . \quad (23)$$

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

$$\xi_i(n) = \left|\frac{u_i(n)}{u_{\max}(n)}\right|. \quad (24)$$

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 θ for T_3 consecutive observations, then the ith hidden unit should be removed, thereby keeping the network size and the computing complexity to minimal.

The above algorithm is used to track the transmitted signal during the first T_2 observation steps of the T_1 time window. Typically the EKF algorithm, i.e., (16) through (21), helps to find a suitable state vector for the reconstructed system, i.e., $\boldsymbol{v}$ in (16). However, once this state vector is found, typically within the T_2 observation steps, this EKF algorithm can be modified to improve convergence of the network by enlarging the attracting domain [5]. Specifically, for (T_2+1) to T_1 observation steps, we replace (18) and (21) by

$$R(n) = \alpha_1 B(\boldsymbol{z}(n))P(n,n-1)B^T(\boldsymbol{z}(n)) + \alpha_2 \quad (25)$$

$$P(n,n) = (I - K(n)B(\boldsymbol{z}(n)))P(n,n-1) \quad (26)$$

where α_1 and α_2 are two parameters used to guarantee the convergence of this algorithm [5].

V Application to Non-Coherent Detection

A Tracking

The Chua's circuit [6] is selected as the chaos generator to be used in the transmitter. This chaos generator can be described by the following dimensionless equations:

$$\begin{aligned} \dot{x}_1 &= \alpha_3(x_2 - \kappa_2(x_1)) \\ \dot{x}_2 &= x_1 - x_2 + x_3 \\ \dot{x}_3 &= \alpha_4(t)x_2 \end{aligned} \quad (27)$$

where α_3 is fixed at 9, and $\alpha_4(t)$ varies according to

$$\alpha_4(t) = -\frac{85}{7} - \frac{15}{7}\sin(t/20), \quad (28)$$

which ensures the chaotic motion of the system. Also, $\kappa_2(.)$ is a piecewise-linear function given by

$$\kappa_2(x_1) = \begin{cases} m_1x_1 + (m_0 - m_1), & x_1 \geq 1 \\ m_0x_1, & |x_1| < 1 \\ m_1x_1 - (m_0 - m_1), & x_1 \leq -1 \end{cases} \quad (29)$$

where $m_0 = -1/7$ and $m_1 = 2/7$. The signal to be transmitted is normalized to the range [−0.5, 0.5], i.e.,

$$s = S(x_3) = \frac{x_3 - x_{3,\min}}{x_{3,\max} - x_{3,\min}} - 0.5 \quad (30)$$

Fig. 2 shows the error waveform of the reconstructed s when SNR is 20dB. It can be noted from Fig. 2 that the error signal has a rapid variation (transient) during the tracking duration T_2, and a variation with a small amplitude at the duration of $[T_2+1, T_1]$, respectively. Thus, we can see clearly that the above algorithm can track the transmitted chaotic signal even when a system parameter varies with time.

B Application to Digital Communication

One useful application of the proposed tracker is to reconstruct the transmitted chaotic signal in a chaos-based communication system. In particular, we consider a simple digital communication system as shown in Fig. 3,

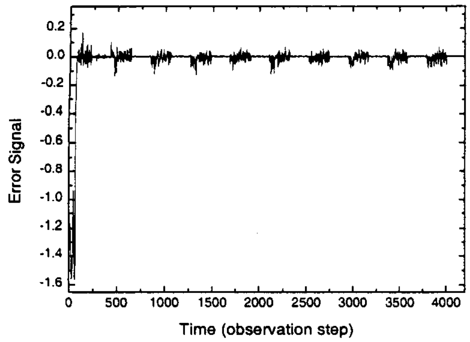

Fig. 2: Error w aveformof the reconstructed signal s when SNR of the channel is 20dB.

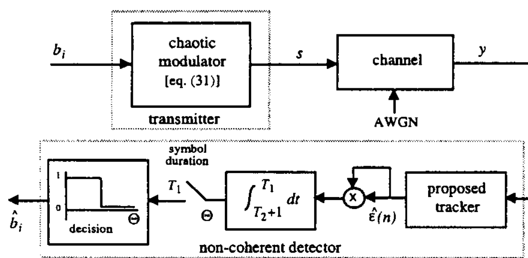

Fig. 3:Bloc k diagram of a chaos-based digital communication system

which effectively employs a chaos-shift-keying modulation. Here, $\{b_i\}$ is the source message bit train which is shown in Fig. 4. The signal to be transmitted is given b y

$$s = \begin{cases} S(x_3), & t \in [1, T_2]; & \text{for all } b_i \\ S(x_3), & t \in [T_2+1, T_1]; & \text{if } b_i = 1 \\ -S(x_3), & t \in [T_2+1, T_1]; & \text{if } b_i = 0\,, \end{cases} \quad (31)$$

for the duration $[1, T_1]$, where $S(.)$ is the normalization function defined in (30) and the unit of t is observation step.

A tthe receiver, an integrator is used to compute the energy of the error signal, Θ, in the interval $T_2 + 1$ to T_1 for eac hT_1 window, i.e.,

$$\Theta = \sum_{n=T_2+1}^{T_1} \hat{\epsilon}^2(n) \quad (32)$$

where $\hat{\epsilon}(n)$ is the trac king error at the nth step. Since the signal s (which is always equal to x_3 during $[1, T_2]$) is trac ked during the firstT_2 observation steps, Θ becomes large if $b_i = 0$, and is small if $b_i = 1$. Thus, Θ can be used b y a suitable decision circuit to determine whidh digital symbol was sen t, i.e.,

$$\hat{b}_i = \begin{cases} 1, & \text{if } \Theta < \Upsilon \\ 0, & \text{if } \Theta > \Upsilon \end{cases} \quad (33)$$

where Υ is the threshold of the decision circuit. Fig. 5 shows the Bit Error Rate (BER) of the retrieved digital message signal for different SNRs in the channel. It can

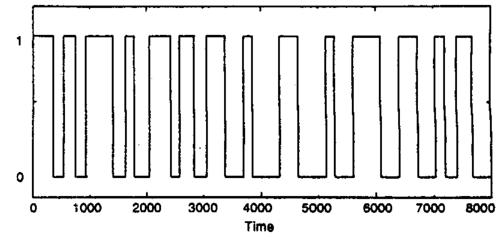

Fig. 4: Source bit stream

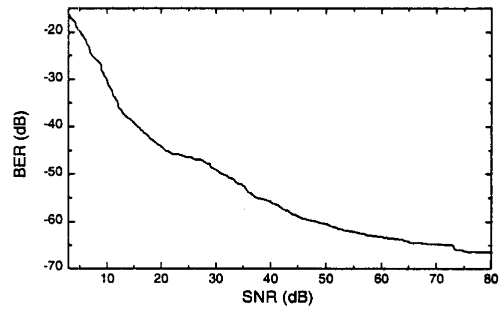

Fig. 5: Bit error rate of the retrieved digital message vs SNR

be seen from Fig. 5 that the digital message signal can be successfully retrieved from an AWGN channel by the proposed demodulator.

VI Conclusion

In this paper, w ehave designed a chaos track er based on a modified RBF neural network. The track er is able to reconstruct a chaotic system which has time-varying parameters and whose observation signal is contaminated by AWGN. A specific application has been demonstrated for a CSK digital communication system employing the proposed track er for non-coherent detection.

References

[1] L.M. Pecora and T.L. Carroll, "Synchronization in chaotic systems", *Phys. Rev. L ett*, Vol. 64, pp. 821–824, 1990.

[2] M.P. Kennedy and G. Kolumbán, "Digital communication using chaos", in *Controlling Chaos and Bifur atio*n*s in Engineering Systems*, G. Chen (Ed.), pp. 477–500, CRC Press, 1999.

[3] F. Tak ens, "Detecting strange attractors in turbulence" in: *Dynamical Systems and Turbulenc e*D. Rand and I. Young (Ed.), pp. 366–381, Berlin: Springer, 1981.

[4] V. Kadirkamanathan and M. Niranjan, "A function estimation approach to sequential learning with neural net- w orks",*Neural Computation*, Vol. 5, pp. 954–975, 1993.

[5] K. Reif and R. Unbehauen, "The extended Kalman filter as an exponential observ er for nonlinear systems",*IEEE T rans. Signal P*r*ocessing*, Vol. 47, pp. 2324–2328, 1999.

[6] R.N. Madan (Ed.), *Chua's Circuit: A Par adigm for Chaos*, Singapore: World Scientific, 1993.