Channel Equalization for Chaos-Based Communication Systems

Jiu-chao FENG*, Chi Kong TSE†, and Francis C. M. LAU†, Nonmembers

SUMMARY A number of schemes have been proposed for communication using chaos over the past years. Regardless of the exact modulation method used, the transmitted signal must go through a physical channel which undesirably introduces distortion to the signal and adds noise to it. The problem is particularly serious when coherent-based demodulation is used because the necessary process of chaos synchronization is difficult to implement in practice. This paper addresses the channel distortion problem and proposes a technique for channel equalization in chaos-based communication systems. The proposed equalization is realized by a modified recurrent neural network (RNN) incorporating a specific training (equalizing) algorithm. Computer simulations are employed to demonstrate the performance of the proposed equalizer in chaotic communication systems. The Hénon map and Chua’s circuit are used to generate chaotic signals. It is shown that the proposed RNN-based equalizer outperforms conventional equalizers.

key words: chaos-based communications, recurrent neural networks, tracking of chaotic signals, channel equalization

1. Introduction

Chaotic signals, by virtue of their wideband and deterministic nature, are well suited for carrying information in a spread-spectrum communication environment [1]–[3]. The wideband feature allows the information to be spread over a wide frequency band, resulting in improved performance in multipath environments and antijamming capability. The deterministic nature allows a high degree of controllability in signal generation and processing. Since 1990’s, the use of chaos in communication has been catalyzed by the pioneering work of Pecora and Carroll [4], [5], who demonstrated the synchronisability of two coupled chaotic systems and hence established the possibility of achieving coherent detection in chaos-based communication. A number of conceptual approaches have been proposed for communication with chaotic signals, e.g., chaotic masking [6], chaos shift key (CSK) [7], etc. Most systems proposed and analyzed are based on the assumption of a rather ideal communication environment, in which signals are transmitted without distortion and with only a moderate amount of added noise. However, in reality, the performance of a communication system can be seriously impaired by channel effects and noise, especially for coherent-type systems where the unfaithful chaos synchronization can make detection rather unreliable [8], [9]. Theoretically, if a wideband chaotic signal is transmitted through a band-limited channel, the inevitable loss of spectral components in the received signal may cause serious problems in the detection. This causes the transmitted signal to spread and overlap over successive time intervals, and this effect is commonly termed inter-symbol interference (ISI) [10]. It has been demonstrated that even simple variations of channel gain or pure phase distortions in the channel may adversely affect the transmitted signal [11], [12]. In addition to linear distortion, the transmitted signal is subject to other contaminations such as thermal noise, impulse noise and nonlinear distortions arising from the modulation process. The idea of channel equalization is to combat the unfavourable channel effects such that the transmitted signal can be preserved with highest integrity [13].

Recently, some approaches have been proposed for combating the effect of channel distortions in chaos-based communication systems. For example, the synchronization-based method [14]–[16] takes advantage of synchronization between the transmitter and receiver to estimate channel distortion. However, it is not easy to choose a suitable coupling parameter (or adaptive coupling parameter) to ensure that all of the conditional Lyapunov exponents in the demodulator are less than zero so that the receiver can approximately synchronize with the transmitter.

Motivated by the lack of effective channel equalization methods for chaos-based communication systems, the present paper attempts to design a channel equalizer for chaotic signals. Linear and nonlinear distortions are considered, in addition to additive white Gaussian noise (AWGN). Specifically, we will employ a modified Recurrent Neural Network (RNN) to realize the equalization task.

This paper is organized as follows. In Sect. 2, we review the background theory of channel equalization in both chaos-based communication systems and conventional communication systems. Details of the proposed equalizing (learning) algorithm are described in Sect. 3.
The simulation performances of the proposed equalizer are demonstrated in Sect.4, where three well-known channel models (linear and nonlinear) will be used to evaluate the performance of the equalizer. Finally, in Sect.5, the performance of the proposed equalizer is compared with that of conventional linear transversal equalizers (LTEs).

2. Preliminaries

Shown in Fig.1 is the block diagram of part of a chaos-based communication system, where *x* is the transmitted signal produced by the chaotic modulator and *h* is the transformation function of the channel. The output of the channel *s* is corrupted by noise *η*, which is usually modelled as an additive white gaussian noise (AWGN) process with a zero mean. At the receiver, the received signal, *y*, first goes through an equalizer which cancels the channel effects and estimates the transmitted signal. In this study we do not consider any specific modulation (demodulation) strategy, and our specific focus here is on cancelling the channel effects.

If the channel *h* is modelled as a linear operator, the output of the channel, *s*, is simply the convolution of the input sequence *x* with *h*, i.e., *s = h*x*. Alternatively, *h* can be modelled as a nonlinear operator, which is given generally by *s = h(x)* [10]. The input to the equalizer is then

\[ y = s + \eta. \] (1)

The problem addressed in this paper may be summarized as follows. Given the noisy and distorted sequence *y*, the problem is to find an equalizer such that the originally transmitted sequence *x* can be reconstructed, or at least a delayed and/or phase shifted version of it. Therefore, the ideal equalization requires \( \hat{x} = \delta_t \delta L e^{j\theta} x \) be achieved, where *t* is time instant, *L* is a time delay, \( \theta \) is a constant phase shift, and \( \delta \) is the Kronecker delta function.

2.1 Conventional Equalizers

The problem of equalization can be interpreted as one of inverse modelling [17], i.e., deconvolving the received sequence in order to reconstruct the original message. The conventional equalizer architecture is shown in Fig.2. The received training sequence, \{\( y(t) \)\}, is filtered by a linear transversal equalizer (LTE) which produces an output for the sample at time *t* − *L* based on \( m \) most recent channel observations made at time *t*, where the integers *m* and *L* are the equalizer’s order (tap) and delay, respectively. In the training stage (Fig.2(a)), some transmitted sample signals are transmitted via the channel, and the received samples are used to train the LTE. Usually, a stochastic gradient algorithm, such as the least mean squares (LMS) algorithm [18], is used to adjust the tap weight \( w_i, i = 0, 1, 2, \cdots, m - 1 \), in the light of the error signal \( e(t - L) \), which is given by

\[ e(t - L) = d(t - L) - \hat{x}(t - L), \] (2)

where \( d(t - L) \) is the desired signal at time *t* − *L*, and \( \hat{x}(t - L) \) is the estimate of \( x(t - L) \). In the equalizing stage (Fig.2(b)), the trained LTE gives the estimate of \( x(t - L) \) from the unknown received sequence when communication commences.

2.2 Reconstruction of Chaotic Signals and Equalization

From the viewpoint of signal processing, the equalization issue of the chaos-based communication systems can be described in terms of signal reconstruction. Specifically, let us consider a chaotic system whose dynamics is governed by the following state equation

\[ \dot{x} = G(x, t) \] (3)

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 \( \mathbb{R}^D \times \mathbb{R} \). When an observer in another space makes

![Fig. 1](image1) Block diagram of part of a chaos-based communication system showing the channel and equalizer.

![Fig. 2](image2) Linear transversal equalizer used in (a) training stage; and (b) equalizing stage.
where $\Psi = [s, s, \ldots, s^{(M-1)}]^T$, where $M \geq (2D+1)$, and $s, \tilde{s}, \ldots, s^{(M-1)}$ denote $\frac{ds}{dt}, \frac{d^2s}{dt^2}, \ldots, \frac{d^{M-1}s}{dt^{M-1}}$, respectively. In other words, there exists a function $\Psi$ such that

$$\mathbf{x} = \Psi(\tilde{s}, t)$$

(5)

where $\Psi = [\psi_1, \psi_2, \ldots, \psi_D]^T$. It should be noted that

$$s = \phi(x(t)) \overset{\text{def}}{=} f_1(x(t)), \tilde{s} = \nabla_x \phi(x(t)) \cdot G(x, t) \overset{\text{def}}{=} f_2(x, t)$$

and

$$s^{(i)} = \frac{ds^{(i-1)}}{dt} = \nabla_x f_i(x, t), G(x, t) \overset{\text{def}}{=} f_{i+1}(x, t)$$

(6)

where $\nabla_x f_i(x, t)$ is the gradient of $f_i(x, t)$ with respect to $x$, and “.” denotes the vector dot (inner) product. Also,

$$\tilde{s} = F(x, t)$$

(7)

where $F = [f_1, f_2, \ldots, f_M]^T$, and $f_i, i = 1, 2, \ldots, M$, is a smooth function. Thus, it is generally possible to reconstruct the chaotic attractor of (3) in a higher dimensional space, if $F(\cdot)$ and $\Psi$ are known. Similar conclusion can be obtained for discrete-time chaotic systems by simply replacing the derivatives with time-advanced state variables, i.e.,

$$(x(t), \dot{x}(t), \cdots) \rightarrow (x(t), x(t+1), \cdots),$$

(8)

where $x$ is one state variable or one observed scalar of a discrete-time chaotic system. However, the receiver usually has no exact knowledge of $F$ and $\Psi$ in practice even when the system is noise-free. Thus, the crucial step is to find $F$ and $\Psi$ under noisy environment in order to realize the reconstruction task. This problem is tackled in this paper by a modified recurrent neural network (RNN) which will be described in the next section.

2.3 RNN and Equalization

The aforementioned equalization problem can be regarded as a nonlinear modelling problem. The nonlinear auto-regressive moving average model (NARMA), which is a widely used tool for modelling nonlinear dynamical system (e.g., in time series processing, nonlinear signal reconstruction, etc.), can be used to describe the said system [20]. Typically, we write

$$x(t) = \varphi(y(t-1), y(t-2), \ldots, y(t-p_1), e(t-1), e(t-2), \ldots, e(t-p_2)) + e(t),$$

(9)

where $e(t)$ is the error signal at time instant $t$ between the original and the estimated signal, $\varphi$ is an unknown function, $p_1$ and $p_2$ are the time delays of the input signal and the error signal, respectively. The conditional mean of $x$ based on the infinite past observations is

$$\hat{x}(t) = E[\varphi(y(t-1), \ldots, y(t-p_1), e(t-1), \ldots, e(t-p_2))|y(t-1), y(t-2), \ldots],$$

(10)

where $E$ denotes expectation. Suppose that the NARMA model is invertible in the sense that there exists a function $\psi$ such that

$$x(t) = \psi(y(t-1), y(t-2), \cdots) + e(t).$$

(11)

Then, given the infinite past observations $y(t-1), y(t-2), \cdots$, one can in principle use the above equation to estimate $e(t-j)$ in (9). In this case the conditional mean estimate is

$$\hat{x}(t) = \varphi(y(t-1), y(t-2), \ldots, y(t-p_1), e(t-1), e(t-2), \ldots, e(t-p_2)).$$

(12)

Since in practice only a finite observation record is available, one cannot perform computation using (12). However, it is possible to approximate (12) by the recursive algorithm [20]

$$\hat{x}(t) = \varphi(y(t-1), y(t-2), \ldots, y(t-p_1), \hat{e}(t-1), \hat{e}(t-2), \ldots, \hat{e}(t-p_2))$$

(13)

where $\hat{e}(j) = x(j) - \hat{x}(j), j = t-1, t-2, \ldots, t-p_2$. The above equation can be approximated by the following recurrent model [20]

$$\hat{x}(t) = \sum_{i=1}^{N} u_i \varphi_i \left\{ \sum_{j=1}^{p_1} w_{ij} y(t-j) + \sum_{j=1}^{p_2} w'_{ij} [x(t-j) - \hat{x}(t-j)] + \theta_i \right\}$$

(14)

which is actually a special case of a general RNN to be described in the following [21]. Here, $u_i$, $w_{ij}$ and $w'_{ij}$ are coefficients and $\theta_i$ is a parameter.

A modified RNN is shown in Fig.3, which is a three-layer network consisting of the input layer, the hidden layer (processing layer) and the output layer. The input vector of the input layer at time instant $t$ is $\mathbf{v}(t)$, which is defined as

$$\mathbf{v}(t) = [v_1(t), \ldots, v_M(t), v_{M+1}(t), \ldots, v_{M+N+1}(t)]^T,$$

(15)
where \( v_i(t), \ 2 \leq i \leq M + 1 \), is the external input which is the delayed version of \( y, \) i.e., \( v_i(t) = y[t - (i - 1)], \) and \( v_i, \ M + 2 \leq i \leq M + N + 1, \) is the feedback input of the \( i \)th input unit at time instant \( t. \) Also, \( N \) is the number of hidden layer units and \( v_1 \) is the bias input which has been fixed at "+1" in this paper.

The internal activity of the \( i \)th hidden unit at time instant \( t \) is given by

\[
r_i(t) = \sum_{j=1}^{M+N+1} w_{ij}(t)v_j(t),
\]

where \( w_{ij}(t) \) is the connection weight between the \( i \)th hidden unit and the \( j \)th input unit at time instant \( t. \) At the next time step \( t + 1, \) the output of the \( i \)th neuron, \( q_i(t + 1), \) is computed using a nonlinear activation function, \( \varphi(\cdot), \) yielding

\[
q_i(t + 1) = \varphi(r_i(t)).
\]

In this study, we choose

\[
\varphi(x) = \tanh(cx),
\]

where \( c \) is constant. Let \( u_i(t) \) be the connection weight between the \( i \)th hidden unit and the output unit. The output of the output unit is given by

\[
\hat{x}(t + 1) = \sum_{i=1}^{N} \left[ u_i(t)q_i(t + 1) \right].
\]

The above estimation procedures, together with the training algorithm described in the following section, can be used to realize the equalization task.

### 3. Training Algorithm

Let \( d(t+1) \) be the desired output of the output unit at time instant \( t + 1. \) The error signal \( e(t + 1) \) is

\[
e(t + 1) = d(t + 1) - \hat{x}(t + 1).
\]

The weight between the hidden layer and the output unit is then updated by a least-mean-square (LMS) algorithm [18], i.e.,

\[
u_i(t + 1) = u_i(t) + \beta_1 e(t + 1)q_i(t + 1),
\]

where \( \beta_1 \) is the learning rate. The instantaneous sum of squared errors of the network is defined as \( e^2(t + 1) = \frac{1}{2} e^2(t + 1). \) Also, we define the local gradient of the \( i \)th hidden unit at time instant \( t + 1, \) \( \gamma_i(t + 1), \) as

\[
\gamma_i(t + 1) = -\frac{\partial e^2(t + 1)}{\partial r_i(t)} = e(t + 1)u_i(t) \frac{\partial q_i(t + 1)}{\partial r_i(t)} = e(t + 1)u_i(t) \varphi'(r_i(t))
\]

where \( \varphi'(\cdot) \) is the derivative of \( \varphi \) with respect to its argument. According to the delta learning law, the weight \( w_{ij} \) \( (i = 1, 2, \ldots, N, \ j = 1, 2, \ldots, M + N + 1) \) can be updated as follows:

\[
w_{ij}^{t}(t + 1) = w_{ij}^{t}(t) + \beta_2 \gamma_i(t + 1) v_j(t),
\]

where \( \beta_2 \) is the learning rate. Now, define the instantaneous sum of squared errors for the hidden layer units as

\[
ev(t) = \frac{1}{2} \sum_{k=1}^{N} e_k^2(t),
\]

where \( e_k(t) \) is the difference (error) in the output of the \( k \)th hidden unit before and after the weight \( w_{ij} \) is updated. Then, the instantaneous weight is updated as

\[
w_{ij}^{t}(t + 1) = w_{ij}^{t}(t) + \beta_3 \frac{\partial e(t)}{\partial w_{ij}^{t}(t + 1)}.
\]

where \( \beta_3 \) is learning rate. From (17) and (24), we have

\[
\frac{\partial e(t)}{\partial w_{ij}^{t}(t + 1)} = \sum_{k=1}^{N} e_k(t) \frac{\partial e_k(t)}{\partial w_{ij}^{t}(t + 1)} = - \sum_{k=1}^{N} e_k(t) \frac{\partial q_k^+(t + 1)}{\partial w_{ij}^{t}(t + 1)},
\]

where \( q_k^+(t + 1) \) is the output of the \( k \)th hidden unit after the weight \( w_{ij} \) is updated to \( w_{ij}^{t}(t + 1). \) To determine the partial derivative \( \partial q_k^+(t + 1)/\partial w_{ij}^{t}(t + 1), \) we differentiate (16) and (17) with respect to \( w_{ij} \). By applying the chain rule, we obtain

\[
\frac{\partial q_k^+(t + 1)}{\partial w_{ij}^{t}(t + 1)} = \frac{\partial q_k^+(t + 1)}{\partial r_k^+(t + 1)} \frac{\partial r_k^+(t + 1)}{\partial w_{ij}^{t}(t + 1)} = \varphi'(r_k^+(t + 1)) \frac{\partial r_k^+(t + 1)}{\partial w_{ij}^{t}(t + 1)},
\]

where \( r_k^+(t + 1) \) corresponds to the updated internal state of hidden unit \( k. \) By using (17), we get

\[
\frac{\partial r_k^-(t + 1)}{\partial w_{ij}^{t}(t + 1)} = \sum_{n=1}^{M+N+1} \frac{\partial}{\partial w_{ij}^{t}(t + 1)} \left[ w_{kn}^{t}(t + 1)v_n(t) \right] = \sum_{n=1}^{M+N+1} \left[ w_{kn}^{t}(t + 1) \frac{\partial v_n(t)}{\partial w_{ij}^{t}(t + 1)} + v_n(t) \frac{\partial w_{kn}^{t}(t + 1)}{\partial w_{ij}^{t}(t + 1)} \right].
\]
Note that the derivative \( \frac{\partial w_{ij}^-(t+1)}{\partial w_{ij}^+(t+1)} \) is equal to one if \( k = i \) and \( n = j \), and is zero otherwise. Thus, we may rewrite (28) as
\[
\frac{\partial r_k^-(t+1)}{\partial w_{ij}^+(t+1)} = \sum_{n=1}^{M+N+1} \left[ w_{kn}^-(t+1) \frac{\partial v_n(t)}{\partial w_{ij}^+(t+1)} \right] + \delta_{ik}v_j(t). \tag{29}
\]
From (15), we have
\[
\frac{\partial v_n(t)}{\partial w_{ij}^+(t+1)} = \begin{cases} \frac{\partial q^+_n(t)}{\partial w_{ij}^+(t+1)} & \text{for } n = M+2, \ldots, M+N+1 \\ 0 & \text{otherwise}. \end{cases} \tag{30}
\]
We may then combine (27), (29) and (30) to yield
\[
\frac{\partial r_k^-(t+1)}{\partial w_{ij}^+(t+1)} = \varphi'(r_k^-(t+1)) \left[ \sum_{n=M+2}^{N+M+1} w_{kn}^-(t+1) \right] \times \frac{\partial q^+_n(t)}{\partial w_{ij}^+(t+1)} + \delta_{ik}v_j(n) \tag{31}\]
where \( \delta \) is the Kronecker delta function. We now define a dynamical system described by a triply indexed set of variables \( \{ \Omega_{ij}^n(t+1) \} \), where
\[
\Omega_{ij}^n(t+1) = \frac{\partial q^+_n(t)}{\partial w_{ij}^+(t+1)}, \tag{32}
\]
where \( j = 1, 2, \ldots, M+N+1, i = 1, 2, \ldots, N, \) and \( n = M+2, \ldots, M+N+1 \). For each time step \( t \) and all appropriate \( n, i \) and \( j \), the dynamics of the system so defined is governed by
\[
\Omega_{ij}^n(t+1) = \varphi'(r_k^-(t+1)) \left[ \sum_{n=M+2}^{N+M+1} w_{kn}^-(t+1) \right] \times \Omega_{ij}^n(t+1) + \delta_{ik}v_j(t). \tag{33}
\]
Finally, the weight between the input layer layer and the hidden layer is updated by
\[
w_{ij}(t+1) = w_{ij}^-(t+1) + \beta_3 \sum_{k=1}^{N} c_k(t+1) \Omega_{ij}^k(t+1). \tag{34}\]

The above procedure is repeatedly applied to all input sample pairs during the training stage.

4. Simulation Study

In this section we simulate a chaos-based communication system which is subject to channel distortion and additive white Gaussian noise. Our purpose is to test the ability of the proposed equalizer in combating the channel effects and noise.

4.1 Chaotic Signal Transmission

Two chaotic systems will be used to evaluate the performance of the proposed equalizer in this paper. The first system is based on the Hénon map:
\[
x_1(t+1) = 1 - \alpha_1 x_1^2(t) + x_2(t)
\]
\[
x_2(t+1) = \alpha_2 x_1(t) \tag{35}
\]
where \( \alpha_1 \) and \( \alpha_2 \) are the bifurcation parameters fixed at 1.4 and 0.3, respectively. In particular, we select \( x_2 \) as the transmitted signal, which guarantees that (5) holds.

The second system is based on the Chua’s circuit [22], which is given by the following dimensionless equations:
\[
\dot{x}_1 = \alpha_3 (x_2 - \kappa_2(x_1))
\]
\[
\dot{x}_2 = x_1 - x_2 + x_3
\]
\[
\dot{x}_3 = \alpha_4 x_2 \tag{36}
\]
where \( \kappa_2(\cdot) \) is a piecewise-linear function given by
\[
\kappa_2(x_1) = \begin{cases} m_1 x_1 + (m_0 - m_1) & \text{for } x_1 \geq 1 \\ m_0 x_1 & \text{for } |x_1| < 1 \\ m_1 x_1 - (m_0 - m_1) & \text{for } x_1 \leq -1 \end{cases} \tag{37}
\]
where \( m_0 = 1/7 \) and \( m_1 = 2/7 \). For different values of \( \alpha_3 \) and \( \alpha_4 \), the system operates in different regimes, e.g., periodic and chaotic regimes. The well-known double scroll Chua’s attractor, for example, is obtained for \( \alpha_3 = 9 \) and \( \alpha_4 = -100/7 \). The attractor has the largest Lyapunov exponent and the Lyapunov dimension equal to 0.23 and 2.13, respectively [23]. In this case, we select \( x_3 \) as the transmitted signal. This choice of the transmitted signal guarantees that the dynamics of the transmission system can be reconstructed, i.e., there exist functions \( F \) and \( \Psi \) such that (5) holds.

4.2 Filtering Effect of Communication Channels

Three channel models will be used to test the performance of the proposed equalizer in this paper. The first two channels are linear channels which can be described in the \( z \) domain by the following transformation functions:
\[
H_1(z) = 1 + 0.5 z^{-1} \quad (\text{Channel I}), \tag{38}
\]
\[
H_2(z) = 0.3 + 0.5 z^{-1} + 0.3 z^{-2} \quad (\text{Channel II}). \tag{39}
\]
These two channel models are widely used to evaluate the performance of equalizers in communication systems [24]. Let us consider the frequency responses of the channels. The amplitude-frequency and phase-frequency responses of the channels are shown in Fig. 4. It is worth noting that from Fig. 4(a), Channel II has a deep spectrum null at an angular frequency of 2.56...
Fig. 4 Frequency responses of Channel I (dotted line) and Channel II (solid line) according to (38) and (39). (a) Magnitude responses; (b) phase responses.

Fig. 5 Nonlinear channel model studied in this paper.

rad/sec, which is difficult to equalize by the usual LTE [10].

The third channel to be studied is a nonlinear channel, which is shown in Fig. 5. This channel is also widely used for testing the performance of equalizers [25]. The model can be described by

\[ y = \tilde{s} + a_1 s^2 + a_2 s^3 + \eta, \]  

where \( a_1 \) and \( a_2 \) are channel parameters which are fixed at 0.2 and −0.1, respectively, and \( \tilde{s} \) is the output of the linear part of the channel which is given by

\[ \tilde{s}(t) = 0.3482 x(t) + 0.8704 x(t-1) + 0.3482 x(t-2). \]  

Thus, the transformation function of the linear component can be expressed as

\[ H(z) = 0.3482 + 0.8704 z^{-1} + 0.3482 z^{-2}. \]  

\[ (40) \]

\[ (41) \]

As an example to illustrate the channel effects, we consider a communication event, in which the transmitted signal \( x_2 \) generated from (35) passes through Channel I. When the signal-to-noise ratio (SNR) is 10 dB, the FFT spectra of the transmitted signal and the received signal are shown in Figs. 6(a) and (b), from which we clearly observe the wideband property of the transmitted signal and the distortion caused by the channel. Furthermore, the return maps reconstructed from the transmitted signal, the distorted signal due to Channel I, and the noisy received signal are shown in Fig. 7. In our previous study [12], it has been shown that without an equalizer, the simple inverse system approach will give unacceptable performance even when the channel, besides AWGN, is an ideal allpass filter, i.e., \( h = 1 \).

4.3 Results

The equalization for each channel model consists of two stages. The first is the training or adaptation stage in which the equalizer makes use of some partially known

Fig. 6 Illustration of channel effects. FFT amplitude spectrum of signal versus frequency for (a) transmitted signal \( x_2 \) from (35), and (b) received signal \( y \) after passing through Channel I and contaminated by noise at SNR = 10 dB.
sampling pairs to adapt to the communication environment. When the training or adaptation is completed, actual communication commences.

In the training stage, 600 training sample pairs are used, each having 7 transmitted signals and 1 known signal at the receiver. Also, noise is added to the training samples at an SNR of 10 dB during the training stage. In the simulation, the RNN is assigned with $M = 7$ and $N = 6$. The training of the equalizer completes when the mean-square-error (MSE) for all samples is less than $10^{-7}$. In this paper, we define the MSE as $\langle (\hat{x} - x)^2 \rangle$, where “$\langle \rangle$” is an averaging operator. Results are summarized as follows:

1. When the RNN-based equalizer is applied to equalize Channels I and II, the following results are obtained:
   - For the Hénon-map system, Fig. 8 shows the MSE of the equalized signals versus the number of iterations in the training duration for Channels I and II. The MSE for each channel in the training stage is averaged over 40 independent realizations in this study. It can be seen from Fig. 8 that the equalizer completes the training in approximately 500 iterations for Channel I and 600 iterations for Channel II. The trained equalizers are then used to test their performances when actual communication takes place. Figure 9 shows the MSE performance of the recovered $\hat{x}_2$ of the Hénon-map system versus SNR for Channel I (solid line) and Channel II (dotted line).
   - For the Chua’s system, Fig. 10 shows the MSE of the training samples versus the number of iterations in the training duration for Channels I and II. When the SNR exceeds 14 dB, the MSE of the equalized signal for the two channels is less than $-80$ dB.
channels I and II. Here, we see that the equalizer completes the training after approximately 500 iterations for Channel I and 700 iterations for Channel II. The trained equalizers are then used to test their performances when actual communication takes place. Figure 11 shows the MSE versus SNR for Channels I and II. When SNR is more than 14 dB, MSE of the equalized signal for the two channels is less than $-76.3$ dB.

2. When the RNN-based equalizer is applied to equalize the nonlinear channel described in Sect. 4.2, the following results are obtained:

- For the Hénon-map system, Fig. 12(a) shows MSE versus the number of iterations during training, and Fig. 12(b) shows the equalization performance of the trained equalizer. We can see that from Fig. 12 the equalizer completes its training after about 838 iterations in the training stage, and the MSE is about $-73.8$ dB when SNR is equal to 14 dB.

- For the Chua’s system, Fig. 13(a) shows MSE versus the number of iterations during training, and Fig. 13(b) shows the equalization performance of the trained equalizer. It can be seen that the equalizer completes its training after about 900 iterations, and the MSE of the equalized signal is approximately $-76.1$ dB when the SNR is equal to 14 dB.

5. Comparison with Other Equalizers

In this section, we compare the proposed RNN-based equalizer with conventional LTEs. In our study, 13-tap and 15-tap LTEs (see Sect. 2.1) are applied to equalize Channels I and II. The results are summarized as follows.

Figures 14(a) and (b) show the MSE performance of the equalized signal versus the SNR of the channel for the communication systems based on the Hénon map and the Chua’s circuit, respectively. When the SNR is 14 dB, the MSE is $-19.2$ dB and $-16.7$ dB for Channels I and II, respectively, for the Hénon map system. Here, the RNN-based equalizer outperforms these LTEs by about 60 dB and 63.3 dB for Channels I and II, respectively. Likewise, for the Chua’s system, the MSE
is about $-32.8$ dB and $-31.8$ dB for Channels I and II, respectively. In this case, the RNN-based equalizer outperforms the LTEs by about $45.5$ dB and $44.5$ dB for Channels I and II, respectively. Also, the LTEs are found completely inadequate for equalizing the nonlinear channel.

6. Conclusions

Channel equalization in chaos-based communication systems has been studied in this paper. The main focus is the kind of channel distortion arising from linear delays as well as nonlinearity. The aim is to derive an effective equalization method such that a transmitted chaotic signal can be preserved with minimal distortion as it passes through a communication channel. Our design approach is based on the use of a recurrent neural network, which has the memory ability to combat dynamical changes, for “probing” the channel characteristics. The equalizer essentially consists of a modified recurrent neural network which incorporates a specific learning algorithm. The Hénon-map system and Chua’s system have been used as chaos generating systems in the simulation study, and three typical linear and nonlinear channels are considered. It has been found that the proposed equalizer can effectively “undo” the channel effects, permitting the chaotic signal to be reconstructed at the receiver.

Finally, it should be noted that the proposed equalization technique has only been tested for some specific linear and nonlinear channel models, which represent basic types of channel distortions. However, in practical communication channels, the types of distortion may arise from a wider range of nonlinear phenomena, which can make the problem much harder to tackle. Future research will therefore focus on tackling real systems with more general distortion characteristics, including bandwidth limitation, existence of colored multiplicative noise, and complex fadings.

Acknowledgment

This work has been supported by a research grant funded by the Hong Kong Polytechnic University.

References


Jiu-chao Feng was born in Sichuan, China. He received the B.S. degree in physics from Southwest China Normal University, Chongqing, China, in 1986, and the M.E. degree in communication and electronic systems from South China University of Technology, Guangzhou, China, in 1997. He is currently working toward the Ph.D. degree at Hong Kong Polytechnic University. Before coming to Hong Kong, he was with the Department of Physics, Southwest China Normal University, Chongqing, China. His areas of interests include digital signal processing, communication theory, nonlinear dynamics and chaos.

Chi Kong Tse received the B.Eng. and Ph.D. degrees in electrical engineering from the University of Melbourne, Australia, in 1987 and 1991, respectively. He is presently a Professor at Hong Kong Polytechnic University, Hong Kong, China. His main research areas include chaotic systems, power electronics and communications. He has published a textbook, over 10 invited book chapters and articles, and over 150 research papers. He also holds a US patent. Since 1999, he has served as associate editor and guest editor for a few international journals including two IEEE Transactions, and has received a few research awards. While with the university, he received twice the President’s Award for Achievements in Research, the Faculty Best Researcher Award, and several other teaching awards.

Francis C. M. Lau received the B.Eng. with first class honours in electrical and electronic engineering and the Ph.D. degree from King's College London, University of London, U.K., in 1989 and 1993, respectively. He is presently an Associate Professor in the Department of Electronic and Information Engineering at Hong Kong Polytechnic University, Hong Kong, China. His main research interests include power control and capacity analysis in mobile communication systems, and chaos-based digital communications.