Digital Signal Processing 62 (2017) 81–90

Contents lists available at ScienceDirect

Digital Signal Processing

www.elsevier.com/locate/dsp

Blind sequence detection using reservoir computing ☆

Xiukai Ruan [a], Chang Li [a,*], Weibo Yang [a], Guihua Cui [a], Haiyong Zhu [a], Zhili Zhou [b], Yuxing Dai [a], Xiaojing Shi [a]

[a] Department of Information and Communication Engineering, Wenzhou University, Wenzhou, 325035, China
[b] School of Information Science and Technology, Sun Yat-sen University, Guangzhou, 510006, China

ARTICLE INFO

Article history:
Available online 2 November 2016

Keywords:
Blind sequence detection (BSD)
Reservoir computing (RC)
Activation function
Support vector regression
Quadrature amplitude modulation (QAM)

ABSTRACT

The performance of *M*-ary quadrature amplitude modulation (QAM) can seriously be degraded by inter-symbol interference (ISI) as the number of levels increases. To mitigate ISI, blind sequence detection (BSD) has very important applications in data transmission systems, particularly where sending a training sequence is disruptive or costly. A new BSD approach of short data in QAM systems using reservoir computing (RC) is presented, together with the detailed theoretical derivation of the algorithm. Its convergence can be guaranteed within a short data packet and, therefore, it works in systems with a much shorter data record and faster time-varying channels. A RC network is constructed to solve the special issue of BSD, with reservoir weight matrix generated via the reduced QR decomposition from the view of receiving signal subspace instead of being selected randomly. The design methods of the activation function and readout function, the variation rule of initial vector which is changed by reservoir weight, and complexity of the proposed algorithm are described, respectively. The readout weight of the RC network is trained and updated by support vector regression (SVR) with a Gaussian insensitive loss function. The correctness and effectiveness of the new approach are verified by simulations, and some special simulation phenomena of the algorithm are discussed.

© 2016 Elsevier Inc. All rights reserved.

1. Introduction

Typical digital communication environments involve transmission of analog pulses over a dispersive medium, inevitably corrupting the received signal by inter-symbol interference (ISI) [1]. ISI is a limiting factor in many communication environments where it causes an irreducible degradation of bit error rate (BER) thus imposing an upper limit on data symbol rate. In addition, the limitations of spectrum allocation have renewed interest in the spectrally efficient quadrature amplitude modulation (QAM) format [2]. High-order QAM modulations (e.g., 64QAM, 128QAM) can increase the available data rate efficiently in a limited channel and spectral efficiency significantly, and it is essential for high-speed far-distance wireless communication. However, the performance of *M*-ary QAM can seriously be degraded by ISI as the number of levels increases. Compared with direct or simple modulation schemes, digital signal processing (DSP) techniques like pre-distortion and channel equalization could be used in high-order modulation systems, and helpful to enhance these systems tolerance. Simultaneously, high-order QAM modulations is still a challenge for high-speed communication due to its complexity and channel distortion [3].

To mitigate ISI, blind equalization has very important applications in data transmission systems, particularly in where sending a training sequence is disruptive or costly [4]. There exists two major blind equalization approaches: the first one estimates channel impulse response and uses an optimum method to recover the transmitted sequence and the second directly equalizes the channel to estimate the transmitted sequence without performing channel estimation. Both of these approaches have to seek a linear filter to minimize a criterion that measures the closeness of the filter output to a discrete-valued signal with the known alphabets. Simultaneously, high-order QAM systems entail a troublesome convergence of the algorithms, and a big data block is absolutely necessary to make these approaches work well.

Motivated by the aforementioned problems, it is therefore an important challenge to construct blind equalization for high-order QAM systems. A new BSD approach using RC [5] was proposed to

☆ This work was partially supported by the National Natural Science Foundation of China (NSFC) (grant nos. 61671329, 61501331, 61303210, 61201426), the Zhejiang Provincial Natural Science Foundation of China (NSFC) (grant nos. LQ16F010010, LY16F010016, LQ16F01001), and the Scientific Research Project of Education Department of Zhejiang Province of China (grant nos. Y201327231, Y201430529).

* Corresponding author.
E-mail addresses: ruanxiukai@yahoo.com (X. Ruan), richare.li@163.com (C. Li), jsj_ywb@wzu.edu.cn (W. Yang), guihua.cui@gmail.com (G. Cui), hyzhu@wzu.edu.cn (H. Zhu), cxzzl@163.com (Z. Zhou), daiyx@wzu.edu.cn (Y. Dai), shi@wzu.edu.cn (X. Shi).

http://dx.doi.org/10.1016/j.dsp.2016.10.012
1051-2004/© 2016 Elsevier Inc. All rights reserved.

Table 1
Notation conventions used in this paper.

Superscript T	Transpose of a matrix
Superscript *	Conjugate of a matrix
Superscript H	Conjugate transpose of a matrix
$diag(a_1, \cdots, a_n)$	Diagonal matrix with diagonal entries $a_1, \cdots$
z^{-1}	Unit-time delay
$range(\mathbf{X})$	The range of a linear transformation $\mathbf{X}$
$span\{\mathbf{x}_1, \cdots, \mathbf{x}_j\}$	Vector space of columns of $\mathbf{x}_1, \cdots, \mathbf{x}_j$
$E\{.\}$	Expectation operation
$\|\mathbf{x}\|_2^2$	Squared 2-norm defined by $\sqrt{\sum \|x_i\|^2}, i = 1, 2, \cdots$
$\nabla_{\mathbf{x}} J(\mathbf{x})$	Derivative of $J(\mathbf{x})$ with respect to the $\mathbf{x}$
$\mathbb{C}^{n\times m}$	The set of $n \times m$ matrices with complex entries
$\mathbb{R}^{n\times m}$	The set of $n \times m$ matrices with real entries
$x_{\mathcal{R}}$	The In-phase (Real) part of x
$x_{\mathcal{I}}$	The Quadrature-phase (Imaginary) part of x

remove ISI of QAM systems. RC is a framework for computation like a recurrent neural network (RNN) that allows for the black box modeling of dynamical systems. RC appeared as a generic name for designing a new research stream including mainly Echo State Networks (ESNs) [6] and Liquid State Machines (LSMs) [7]. Together they appeared under the umbrella term of RC: both approaches provide a reservoir with computational complexity that can be harnessed to solve a variety of nonlinear problems. These two methods built-in many highly challenging ideas which toward a new computational paradigm of neural networks, have quickly made RC become popular.

In this work, we focus the study on BSD using RC technique to remove ISI of QAM systems even if the training sequence is either too short or absent. It is noteworthy that the proposed approach could not be classified as one of the above two types of blind equalization techniques since it can estimate directly the input sequence without getting the equalizer coefficients and estimating channel impulse response at the receiver. In addition, the proposed approach guarantees a convergence within a short data packet, therefore, can work in systems with a much shorter data record and faster time-varying channels.

The rest of this paper is organized as follows. Section 2 presents the QAM and BSD problem formulation. Next, in Section 3, a new BSD approach along with the main theoretical results are described. A RC network is constructed to solve the special issue of BSD, with reservoir weight matrix generated via the reduced QR decomposition from the view of receiving signal subspace instead of selected randomly. And the variation rule of initial vector which is changed by reservoir weight is shown. Meantime, the activation function and readout function of this RC network are designed, respectively. In Section 4, the readout weight of the RC network is trained and updated by support vector regression (SVR) [8] with a Gaussian insensitive loss function and the complexity of the proposed algorithm is illustrated. In Section 5, simulation results are represented to verify the correctness and effectiveness of the new approach, and some special simulation phenomena of the algorithm are demonstrated, followed by conclusion and discussion in Section 6.

Throughout the paper, bold letters denote matrices or vectors, and the notation, shown in Table 1, is used.

2. QAM concept and BSD problem

QAM is widely used in many digital data radio and optical coherent communications. For domestic broadcast applications for example, high-order modulations (e.g., 64QAM) are often used in digital cable television and cable modem applications [11]. A variety of QAM are available and the most common one is M-QAM, $M = 4, 8, \cdots, 64$. By moving to higher-order constellations, it is possible to transmit more bits per symbol, which reduces bandwidth. However, the performance of QAM can seriously be degraded due to ISI as the number of levels increases. A typical BSD setup is depicted in Fig. 1 using a simple system diagram. The complex baseband model for a QAM data communication system consists of an unknown linear time-invariant channel $\{h\}$. The transmitter generates a sequence of complex-valued random input data $\{s_N\}$, each element of which belongs to a complex alphabet of QAM symbols. The data sequence $\{s_N\}$ is sent through a complex channel whose output x_N is observed by the receiver. The function of BSD at the receiver is to estimate the delayed version of original data $\{s_N\}$, $\{s_{N-k}\}$, from the received signal x_N (where k is bulk delay).

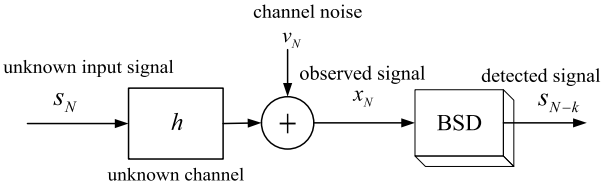

Fig. 1. Baseband communication system model.

This system identification scenario is a linear single-input multiple-output (SIMO) multichannel model which can always be treated as a collection of single-input single-output (SISO) models. It is well known that fractionally spaced samples of a single baseband received signal lead to a SIMO model. In order to simplify the presentation of the proposed sequence estimation method, without loss of generality, in a noise-free environment, a SIMO channel system whose i-th sub-channel output is

$$x_i(t) = \sum_{\tau=0}^{L_h-1} h_i(\tau)s(t-\tau) \tag{1}$$

where $s(t)$ is the QAM input signal and $h_i(\tau)$ is the i-th channel impulse response with length of L_h. Clearly, the column vector $\mathbf{x}(t)$ can be expressed as

$$\mathbf{x}(t) = \left[\mathbf{h}_1, \mathbf{h}_2, \cdots, \mathbf{h}_q\right]\mathbf{s}(t) \tag{2}$$

where $\mathbf{x}(t) \in \mathbb{C}^{q\times 1}$, q denotes the over-sample factor. And

$$\mathbf{h}_p = \left[h_p(0), h_p(1), \cdots, h_p(L_h-1)\right], p = 1, 2, \cdots, q \tag{3}$$

and

$$\mathbf{s}(t) = [s(t), s(t-1), \cdots, s(t-L_h+1)]^T \tag{4}$$

Thus the received data matrix can be formulated as

$$\mathbf{X} = \mathbf{S}\boldsymbol{\Gamma} \tag{5}$$

where

$$\mathbf{S} = [\mathbf{s}_N(t), \mathbf{s}_N(t-1), \cdots, \mathbf{s}_N(t-L_h-L_w)] \in \mathbb{C}^{N\times(L_h+L_w+1)}, \tag{6}$$

and

$$\mathbf{s}_N(t) = [s(t), s(t-1), \cdots, s(t-N+1)]^T \tag{7}$$

is the transmitted signal matrix; N is the source signal length; and $\boldsymbol{\Gamma} = \left[\mathbf{H}_1, \mathbf{H}_2, \cdots, \mathbf{H}_q\right]$ with

$$\mathbf{H}_p = \begin{bmatrix} \mathbf{h}_p & 0 & \cdots & 0 \\ 0 & \mathbf{h}_p & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \mathbf{h}_p \end{bmatrix}, p = 1, 2, \cdots, q \tag{8}$$

where $\mathbf{H}_p \in \mathbb{C}^{(L_w+1)\times(L_h+L_w+1)}$.

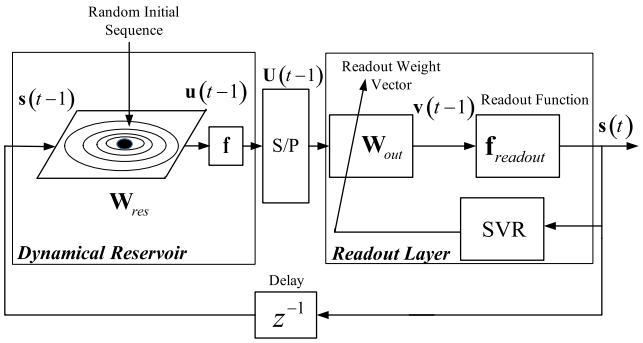

Fig. 2. Schematic overview of the approach of BSD using RC.

We adopt the following basic assumptions:

1. Channel order L_h is assumed to be known as a priori.
2. The input signal $\mathbf{s}$ is zero-mean and temporally i.i.d.
3. Additive noise is spatially and temporally white noise and is statistically independent of the source.

Without training sequence, and without estimating the channel impulse response or getting the equalizer coefficients, our objective is to detect the source sequence $\mathbf{s}_N(t)$ directly from the observed received data matrix $\mathbf{X}$ by utilizing RC.

3. BSD using RC

3.1. The network structure of RC for BSD

In this section, the mathematical model of BSD based on RC is given out. A traditional reservoir is a dynamical system that is composed of a pool of recurrently connected nonlinear computational nodes. The basic idea behind RC is to drive a randomly created RNN (the reservoir, remains unchanged during training) with the task input signal, and from the input-excited RNN-internal dynamics distil a desired output signal by a trainable readout mechanism (often just a linear readout trained) by linear regression of the target output on the excited internal activation traces. The RC approach is a simpler and more straightforward way to train RNNs as opposed to techniques such as back-propagation (BP) through time. This was traditionally one of the difficulties faced when working with RNNs. It is passively excited by the input signal and maintained of its state a nonlinear transformation in the input history [12,13]. To address the special issue of BSD, we construct a network structure of RC as shown by Fig. 2.

The network structure consists of two main components and operates as follows: firstly, the network starts from a random initial vector. The random initial vector is imported here as an external field with which the RC coupled. The input sequence is coupled with the reservoir weight $\mathbf{W}_{res}$ for only a single time step and then the network dynamics are left to run. Then the output sequence of dynamical reservoir is transformed into an output sequence via a nonlinear activation function $\mathbf{f}(.)$. The output sequence is concatenated to construct a vector representing the input source sequence. These are then fed into the second component of the system, a feed forward neural network that is trained using the standard SVR algorithm. The structure of the RC is fixed and if large enough, the network is capable of generating a wide variety of dynamics. Here, as the input data are presented to the network, a memory trace is formed within the RC network because its feedback connections create an internal memory. In Fig. 2, S/P denotes serial to parallel conversion.

Due to its recurrent connections, a reservoir can capture long-term dynamics of the input data and is therefore an appealing method to create feature vectors that are determined by the recent past. This RC network is a discrete-time, discrete-value real-time system model with the following state update equations

$$\mathbf{u}(t-1) = \mathbf{f}(\mathbf{W}_{res}\mathbf{s}(t-1)) \tag{9}$$

$$\mathbf{U}(t-1) = [\mathbf{u}_0, \mathbf{u}_1, \cdots, \mathbf{u}_N] \tag{10}$$

$$\mathbf{u}_i = \left[u_i, u_{i+1}, \cdots, u_{i-L_w+1}\right], i = 0, 1, \cdots, N-1 \tag{11}$$

$$\mathbf{s}(t) = \mathbf{f}_{readout}(\mathbf{U}(t-1)\mathbf{w}_{out}(t-1)) \tag{12}$$

where $\mathbf{w}_{out}(t-1) = \left[w_0, w_1, \cdots, w_{L_w-1}\right]^T \in \mathbb{C}^{L_w \times 1}$ denotes the readout weight, $\mathbf{f}(.)$ is the activation function operator mapping from $\mathbb{C}^{N\times 1} \to \mathbb{C}^{N\times 1}$, $\mathbf{u}(t)$ the output of reservoir, $\mathbf{f}_{readout}(.)$ the readout function mapping from $\mathbb{C}^{N\times 1} \to \mathbb{C}^{N\times 1}$, $\mathbf{U}(t) \in \mathbb{C}^{N\times L_w}$, $\mathbf{s}(t)$ the output of the network at a time step t.

3.2. Reservoir weight matrix

In RC, there is no need to train the reservoir weight but the readout weight of the network; this allows a fast parallel implementation for large reservoirs. However it is necessary to adjust parameters to create a "good" reservoir for a given application. It seems obvious that when addressing a specific modeling task, a specific design of reservoir adapted to the task will lead to better results than that of a naive random creation [13]. Several solutions, such as general guideline, reservoir adaptation and reservoir pruning [14], have been proposed to construct a reservoir. In this paper, the reservoir weight matrix is devised via the reduced QR decomposition [15,16] from the view of receiving signal subspace.

Since $\mathbf{X} \in \mathbb{C}^{N\times(L_w+1)q}$ with $N \geq (L_w+1)q$, we can apply the reduced QR decomposition to $\mathbf{X}$

$$\mathbf{X} = \hat{\mathbf{Q}}\hat{\mathbf{R}} \tag{13}$$

where $\hat{\mathbf{Q}} \in \mathbb{C}^{N\times(L_w+1)q}$ is a matrix with orthonormal columns and $\hat{\mathbf{R}} \in \mathbb{C}^{(L_w+1)q\times(L_w+1)q}$ is an upper triangular matrix such that $\hat{\mathbf{R}}(j,j) \neq 0$, $j = 1, \cdots, (L_w+1)q$.

Here, $\hat{\mathbf{Q}}$ provides an orthonormal basis for $range(\mathbf{X})$, i.e., the columns of $\mathbf{X}$ are linear combinations of the columns of $\hat{\mathbf{Q}}$. In fact, we have $range(\hat{\mathbf{Q}}) = range(\mathbf{X})$. This is true since $\mathbf{X}\mathbf{y} = \hat{\mathbf{Q}}\hat{\mathbf{R}}\mathbf{y} = \hat{\mathbf{Q}}\mathbf{z}$ for some $\mathbf{z}$ so that $range(\mathbf{X}) \subseteq range(\hat{\mathbf{Q}})$. Moreover, $range(\hat{\mathbf{Q}}) \subseteq range(\mathbf{X})$ since we can write $\mathbf{X}\hat{\mathbf{R}}^{-1} = \hat{\mathbf{Q}}$ because $\hat{\mathbf{R}}$ is upper triangular with nonzero diagonal elements. (Now we have $\hat{\mathbf{Q}}\mathbf{y} = \mathbf{X}\hat{\mathbf{R}}^{-1}\mathbf{y} = \mathbf{X}\mathbf{z}$ for some $\mathbf{z}$). Note that any partial sets of columns satisfy the same property, i.e, $span\{\mathbf{x}_1, \cdots, \mathbf{x}_j\} = span\{\mathbf{q}_1, \cdots, \mathbf{q}_j\}$, $j = 1, \cdots, (L_w+1)q$.

There is a well-defined gap in the singular values of $\mathbf{X}$, in other words, it exists an index r that $\sigma(r+1) \ll \sigma(r)$, where $\sigma(.)$ denotes the singular of $\mathbf{X}$, then the subset selection will tend to produce a subset containing the most important columns (rules) of $\mathbf{X}$ [15,16]. Hence, $\hat{\mathbf{Q}}$ can be partitioned into two block matrices $\mathbf{Q}$ and $\mathbf{Q}_c$, where $\mathbf{Q} \in \mathbb{C}^{N\times r}, r \leq (L_w+1)q$.

Thus the RC matrix $\bar{\mathbf{W}}_{res}$ can be formulated as

$$\bar{\mathbf{W}}_{res} = \mathbf{Q}\mathbf{Q}^H. \tag{14}$$

Clearly, $\bar{\mathbf{W}}_{res}$ is an Hermitian matrix.

Definition 1. An $n \times n$ square matrix $\mathbf{A}$ is idempotent if and only if $\mathbf{A}^2 \equiv \mathbf{A} = \mathbf{A}$ [16].

Theorem 1. *The RC weight matrix $\mathbf{W}_{res}$ is a non-negative definite idempotent matrix, its eigenvalues are either 0 or 1 and its spectral radius $\rho(\mathbf{W}_{res})$ is 1. (The proof of Theorem 1 appears in the Appendix A.)*

The spectral radius $\rho(\mathbf{W}_{res})$ should be close to, but not equal to unity for a reservoir to work well [17]. Larger $\rho(\mathbf{W}_{res})$ has the effect of driving signals $\mathbf{s}(t)$ into more nonlinear regions of activation function units (further from 0) [18]. $\rho(\mathbf{W}_{res})$ can be interpreted as the forgetting factor of the RC networks as it determines the extent to which the network forgets its previous inputs. To achieve this goal, a magnification λ is introduced, leads to the RC matrix be

$$\mathbf{W}_{res} = \lambda \bar{\mathbf{W}}_{res}^{H} = \lambda \mathbf{Q}\mathbf{Q}^{H}. \tag{15}$$

3.3. The activation function for the output of reservoir weight

Both the real and imaginary parts of QAM signals belong to a discrete and finite alphabetical set $A = \{\pm 1, \pm 3, \pm 5, \cdots\}$, which is an important priori information and extremely useful, hence we only discuss the real part here. Using the information with the idea of [19], the following RC activation function f is obtained as $f = f_{\mathcal{R}}(x, N_s) + f_{\mathcal{I}}(x, N_s)$, where

$$f_{\mathcal{R}}(x, N_s) = f_{\mathcal{I}}(x, N_s) = \sum_{i=1}^{N_s - 1} \left[\frac{2}{1 + e^{-a(x+b_i)}} \right] - (N_s - 1) \tag{16}$$

where $b_i = N_s - 2i, i = 1, 2, \cdots, (N_s - 1)$ are the unstable inflexion points of the activation function, a is the attenuation factor of the activation function and N_s denotes the number of multilevels (e.g., $N_s = 8$ for 64QAM).

The value of a influences the degree of steepness of the $f_{\mathcal{R}}$ curve, and the convergence rate and performance of the algorithm significantly. If a takes a small value, the f function will expedite the RC output signals adjusting proceedings, in other words, the convergence rate is accelerated, but a excessive residual error remained even if the algorithm is convergent. On the contrary, with a large of a, the convergence rate will decrease, but a better performance in return is achieved.

3.4. The magnification λ

Now, we will illustrate another important function of the magnification λ. To show the function of λ clearly, we assume $\lambda = 1$, in other words, $\mathbf{W}_{res} = \mathbf{Q}\mathbf{Q}^{H}$. Before the RC network becomes steady, there exists a *Transient Process* which is related to the initial state vector. It means the convergence curve of the algorithm will be rolling obviously. Hence, it is necessary to study the change law of the initial state vector when RC weight matrix $\mathbf{W}_{res}$ begins to work.

Without loss of generality, we only consider the case of no-activation function. Then, we note the initial state vector $\mathbf{s}(0) \in A$, where $A = \{\pm 1, \pm 3, \cdots, \pm d_n | d_n = 2n - 1\}, n = 1, 2, \cdots$ is the finite alphabet set for QAM signal, $(2n)$ is the total number of the elements in the finite alphabet set.

In the initial step, $\mathbf{u}(0) = \mathbf{W}_{res}\mathbf{s}(0)$, $u_i(0) = \sum_{j=1}^{N} w_{ij} s_j(0)$, $i = 1, 2, \cdots, N$. For 64QAM case, since $\mathbf{s}(0) \in A$, and the alphabets are equiprobable, symmetric about the origin, and independent, the mean value and the standard deviation of $s_i(0), i = 1, 2, \cdots, N$, can be calculated by

$$\begin{aligned} m_{si} &= E\{s_i(0)\} \\ &= \frac{1}{8} \sum_{l \in \pm 1, \pm 3, \pm 5, \pm 7} l = \frac{1}{8} [(-7) + (-5) + \cdots + 7] = 0 \end{aligned} \tag{17}$$

and

$$\begin{aligned} \delta_{si}^2 &= E\left\{ [s_i(0) - m_{si}]^2 \right\} = E\left\{ [s_i(0)]^2 \right\} \\ &= \sum_{l \in \pm 1, \pm 3, \pm 5, \pm 7} l^2 = 21 \approx (4.58)^2, \quad i = 1, 2, \cdots, N, \end{aligned} \tag{18}$$

respectively.

Hence, we can obtain the mean value and the standard deviation of $u_i(0), i = 1, 2, \cdots, N$ that

$$\begin{aligned} m_{ui} &= E\{u_i(0)\} \\ &= \sum_{j=1}^{N} E\left\{ w_{ij} s_j(0) \right\} = \sum_{j=1}^{N} w_{ij} E\left\{ s_j(0) \right\} = 0, \end{aligned} \tag{19}$$

$$\begin{aligned} \delta_{ui}^2 &= E\left\{ [u_i(0) - m_{ui}]^2 \right\} = \sum_{j=1}^{N} E\left\{ [w_{ij} s_j(0)]^2 \right\} \\ &= \delta_j^2 \sum_{j=1}^{N} w_{ij}^2 = \delta_{si}^2 \|\mathbf{w}_i\|_2^2, \quad i = 1, 2, \cdots, N. \end{aligned} \tag{20}$$

Then we get $u_i \sim \mathcal{N}(0, \delta_{ui}^2)$, where $\delta_{ui}^2 = \delta_{si}^2 \|\mathbf{w}_i\|_2^2 \approx 4.58^2 \|\mathbf{w}_i\|_2^2$, $i = 1, 2, \cdots, N$.

Since $\mathbf{W}_{res}$ is the projection operator of signal subspace, the ratio of the size of signal subspace to the size of signal space can be calculated by $\mu = (L_w + L_h + 1)/N$. Considering $N \gg (L_w + L_h + 1)$, we have $\mu \ll 1$. Hence, by utilizing "3σ *Rule*" [20] of normal distribution, it is true that the probability $P\left(u_i(0) \in (-3\|\mathbf{w}_i\|\delta_{ui}^2, 3\|\mathbf{w}_i\|\delta_{ui}^2\right] = 99.7\%$, for $\forall i = 1, 2, \cdots, N$. For example, if $L_w = 10, L_h = 5$, and $N = 1500$, then $\mu = 0.01$, so it can be estimated that $\|\mathbf{w}_i\|_2^2 \approx 0.01$, in other words, $u_i \sim \mathcal{N}(0, 0.458^2)$. It is true that $P(-1.375 < u_i(0) \leq 1.375) = 99.7\%$, for $\forall i = 1, 2, \cdots, N$. It means that, in the initial step, $\mathbf{W}_{res}$ will make the signal points of input sequence flock around the range of $(-1.375, 1.375]$. However, we hope that signal points are distributed evenly in the range of $[-7, 7]$. Thus, a magnification λ whose value is slightly greater than 1 is needed to amplify the input sequence during iteration.

4. Readout weight training procedure and analysis

4.1. Readout weight training using SVR

The output layer of the reservoir, mapping the liquid state into an output, consists of a set of neurons connected to the reservoir. As mentioned previously, only synaptic weights from the reservoir to the output layer will be subject to the training procedure. In other words, the training process of RC network is equivalent to the problem solving process of readout weight vector. The reservoir can be considered as a temporal kernel, and the standard linear readout vector $\mathbf{w}_{out}$ can be trained using loss functions and regularization as in SVR [8–10]. An extra term, dependent on the readout weight $\mathbf{w}_{out}$ is added to the regression cost function

$$\hat{\mathbf{w}}_{out} = \arg\min \left\{ \frac{1}{2} \|\mathbf{w}_{out}\|_2^2 + C \cdot E\left\{ L_\delta\left(R_p, |\mathbf{u}_i^T \mathbf{w}_{out}|^2 \right) \right\} \right\} \tag{21}$$

where $R_p = \frac{E\{s_R^4\}}{E\{s_R^2\}}$, $\mathbf{u}_i = \left[u_i, u_{i-1}, \cdots, u_{i-L_w+1}\right]^T$, $i = L_w - 1, L_w, \cdots, N - 1$, the value of δ is a parameter that controls the amount of shrink age. $L_\delta\left(R_p, |\mathbf{u}_i^T \mathbf{w}_{out}|^2\right)$ is Loss Function. Here, we only discuss the case that $L_\delta\left(R_p, |\mathbf{u}\mathbf{w}_{out}|\right)$ is an Gaussian δ-insensitive loss function

$$\begin{aligned} L_\delta\left(R_p, |\mathbf{u}_i^T \mathbf{w}_{out}|^2\right) &= \left| |\mathbf{u}_i^T \mathbf{w}_{out}|^2 - R_p \right|_\delta^2 \\ &= \begin{cases} \left(|\mathbf{u}_i^T \mathbf{w}_{out}|^2 - R_p - \delta\right)^2, & \text{if } \left||\mathbf{u}_i^T \mathbf{w}_{out}|^2 - R_p\right| \geq \delta \\ 0, & \text{if } \left||\mathbf{u}_i^T \mathbf{w}_{out}|^2 - R_p\right| < \delta \end{cases} \end{aligned} \tag{22}$$

Then, the cost function of (21) can be rewritten as

$$\hat{\mathbf{w}}_{out} = \arg\min \left\{ \frac{1}{2}\|\mathbf{w}_{out}\|_2^2 + \frac{C}{N}\sum_{i=L_w-1}^{N} L_\delta\left(\left||\mathbf{u}_i^T\mathbf{w}_{out}|^2 - R_p\right|^2\right)\right\}. \quad (23)$$

By defining $J(\mathbf{w}_{out}) = \frac{1}{2}\|\mathbf{w}_{out}\|_2^2 + \frac{C}{N}\sum_{i=L_w-1}^{N} L_\delta \times \left(\left||\mathbf{u}_i^T\mathbf{w}_{out}|^2 - R_p\right|^2\right)$, then we have

$$\nabla_{\mathbf{w}_{out}} J(\mathbf{w}_{out}) = \mathbf{w}_{out} + \underbrace{\frac{C}{N}\sum_{i=L-1}^{N}\left\{2A_i y_i^* y_i \mathbf{u}_i^T\mathbf{w}_{out}\mathbf{u}_i^*\right\}}_{III} - \underbrace{\frac{C}{N}\sum_{i=L_w-1}^{N}\left\{(2A_i R_p + B_i\upsilon_i^l)y_i\mathbf{u}_i^*\right\}}_{IV}. \quad (24)$$

where $A_i = \frac{1}{2}\left(\frac{1}{\upsilon_i^l}\frac{dL_\delta(\upsilon)}{d\upsilon}\Big|_{\upsilon_i^l} + \frac{dL_\delta^2(\upsilon)}{d\upsilon^2}\Big|_{\upsilon_i^l}\right)$ and $B_i = \frac{dL_\delta^2(\upsilon)}{d\upsilon^2}\Big|_{\upsilon_i^l}$. Obviously, both A_i and B_i are constant for all i.

Next, we will express this equation in matrix notation. For simplicity, it is noted that the second term of (24) as *III*, the third as *IV*. Hence, the *III* term of (24) can be formulated as (25) in matrix notation (see Appendix B).

$$III = \mathbf{U}^H\mathbf{A}\mathbf{Y}\mathbf{U}\mathbf{w}_{out}. \quad (25)$$

Similarly, we note that $\mathbf{B} = diag(B_{L_w-1}, B_{L_w}, \cdots, B_N) \in \mathbb{C}^{(N-L_w+1)\times(N-L_w+1)}$, $\mathbf{E} = diag(\upsilon_{L_w-1}^l, \upsilon_{L_w}^l, \cdots, \upsilon_N^l) \in \mathbb{C}^{(N-L_w+1)\times(N-L_w+1)}$ and $\mathbf{y} = \left[y_{L_w-1}, y_{L_w}, \cdots, y_N\right]^T \in \mathbb{C}^{(N-L_w+1)\times 1}$. Then the IV term can be expressed in matrix notation

$$IV = \mathbf{U}^H(2R_p\mathbf{A} + \mathbf{B}\mathbf{E})\mathbf{y}. \quad (26)$$

Hence we obtain

$$\nabla_{\mathbf{w}_{out}} J(\mathbf{w}_{out}) = \left(\frac{C}{N}\mathbf{U}^H\mathbf{A}\mathbf{Y}\mathbf{U} + \mathbf{I}\right)\mathbf{w}_{out} - \mathbf{U}^H(2R_p\mathbf{A} + \mathbf{B}\mathbf{E})\mathbf{y} \quad (27)$$

where $\mathbf{I}$ is an identity matrix. By setting $\nabla_{\mathbf{w}_{out}} J(\mathbf{w}_{out}) = 0$ and combined with

$$\frac{\partial^2 D(\mathbf{w}_{out})}{\partial\mathbf{w}_{out}\partial\mathbf{w}_{out}^H} = \frac{C}{N}\mathbf{U}^H\mathbf{A}\mathbf{Y}\mathbf{U} + \mathbf{I}, \quad (28)$$

and assuming $(\frac{C}{N}\mathbf{U}^H\mathbf{A}\mathbf{Y}\mathbf{U} + \mathbf{I})$ to be full column rank, the solution can be obtained

$$\hat{\mathbf{w}}_{out} = \left(\frac{C}{N}\mathbf{U}^H\mathbf{A}\mathbf{Y}\mathbf{U} + \mathbf{I}\right)^{-1} - \mathbf{U}^H(2R_p\mathbf{A} + \mathbf{B}\mathbf{E})\mathbf{y}. \quad (29)$$

Hence, the output of RC network is

$$\tilde{\mathbf{y}} = \mathbf{f}_{readout}\left(\mathbf{U}\hat{\mathbf{w}}_{out}\right) = \mathbf{f}_{readout}\left(\left(\frac{C}{N}\mathbf{U}^H\mathbf{A}\mathbf{Y}\mathbf{U} + \mathbf{I}\right)^{-1} - \mathbf{U}^H(2R_p\mathbf{A} + \mathbf{B}\mathbf{E})\mathbf{y}\right) \quad (30)$$

where $\mathbf{f}_{readout}(\cdot)$ is the readout function that can be linear or nonlinear.

4.2. Readout function

In traditional reservoir systems, the readout nodes are linear, which is suitable to BSD issues, we adopt the following readout function to improve the convergence precision

$$f_{readout}(y) = g(y) + jg(y), g(y) = y + \gamma\sin(\pi y) \quad (31)$$

where $\gamma \in (0, \frac{1}{\pi}]$. Nonlinear nodes call for an iterative training and the training time will be proportional to the number of iterations that are required to converge to a good solution [18].

4.3. Computational complexity analysis

One advantage of an iteration algorithm is its simple update rule, making it suitable for hardware-only implementations. However, the proposed RC algorithm requires a matrix inversion and needs to update and store several matrices at each iteration thus leading to a high computational burden and more memory space to store its matrices. Overall, RC approach needs about $(5L_wN_{RC}^2 + 2N_{RC}L_w^2 + L_wN_{RC})$ multiplication operations per iteration. In addition, it needs to calculate the inversion of an $L_w \times L_w$ matrix per iteration. We denote the number of iterations as K_{RC}. Due to $N_{RC} \gg L_w$, the overall computational complexity includes $(5L_wK_{RC}N_{RC}^2)$ multiplication operations, K_{RC} times of the inversion operation of an $L_w \times L_w$ matrix. Next, we compare the complexity of the proposed algorithm with other BSD algorithms. It should be emphasized that the length of data $N_{RC} < 1500$, the length readout weight of $L_w < 20$, and the number of iterations $K_{RC} < 100$ for 64QAM, generally.

For simplicity, we denote the length of data, the length of equalizer and the number of iterations of the Multi-module algorithm (MMA) approach [21,22] as N_{MMA}, L_{MMA} and K_{MMA}, respectively; and denote the length of data and the length of equalizer, and the number of iterations of the super-exponential (SE) algorithm [23] as N_{SE}, L_{SE} and K_{SE}, respectively.

- MMA algorithm, requires $N_{MMA}L_{MMA}K_{MMA}$ multiplication operations. Here $K_{MMA} \gg K_{RC}$ and N_{MMA} is extremely larger than N_{RC} in general.
- The SE algorithm requires $N_{SE}L_{SE}K_{SE}$ operations for output computation, $6N_{SE}L_{SE}K_{SE}$ operations for the computation of the empirical cumulate expression and $L_{SE}^2K_{SE}$ operations for the multiplication with the inverse covariance matrix. Similarly, $K_{SE} \gg K_{RC}$ and N_{SE} is also extremely larger than N_{RC}.

5. Simulation and results

In this section, we provide simulation results to illustrate and verify the theory developed. Unless noted otherwise, our experiments are based on a complex SIMO multipath channel [24]. A raised-cosine pulse $P(t)$ limited in $6T$ with roll-off factor 0.10 and a two-ray multipath channel $c(t) = \delta(t) - 0.7\delta(t - T/3)$ which results in the overall channel impulse response of $h_{\mathcal{R}}(t) = h_{\mathcal{I}}(t) = c(t) \otimes P(t) = P(t) - 0.7P(t - T/3)$. The source QAM sequence $\mathbf{s}(t)$ which is i.i.d. and satisfies block-fading feature, is generated randomly for one independent trial; the over sampling factor is $q = 3$, and the sub-channel order is $L_h = 4$. The noise is zero mean, white, and Gaussian. Meanwhile, we adopt a Gaussian δ-insensitive quadratic loss function with $\delta = 1$. The magnification of reservoir weight $\mathbf{w}_{out}$ is $\lambda = 1.03$, the attenuation factor of the RC activation function $a = 10$, and the factor γ of readout function $f_{readout}$ is 0.3, respectively. One initial sequence whose real and imaginary part belong to a discrete and finite alphabetical set $A = \{\pm1, \pm3, \pm5, \pm7\}$ is generated randomly to drive the RC network at the beginning of iteration for any independent trial. All results are averaged over 500 Monte Carlo runs.

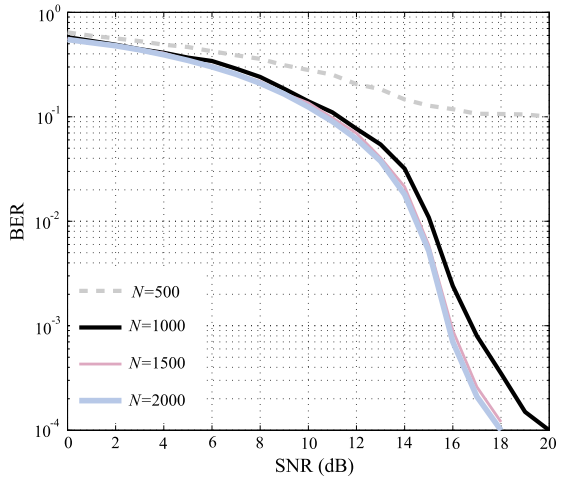

Fig. 3. 64QAM case, the average BER of BSD using RC approach with the length of readout weight $L_w = 9$.

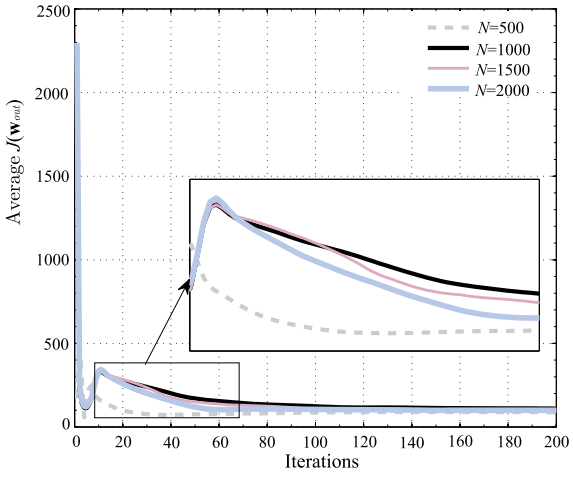

Fig. 4. 64QAM case, the curves of the average cost function $J(\mathbf{w}_{out})$ of the RC approach with the length of readout weight $L_w = 9$.

5.1. The experimental phenomena of RC approach

Fig. 3 shows the average BER curves of BSD using the RC approach with the length of readout weight $L_w = 9$ when data length N (= 500, 1000, 1500, 2000), respectively. It is clear that this improves performance of the lengthened length of the received data. However, it is shown that the improvement will be limited when N is long enough (e.g., $N \geq 1500$). As we all know that increasing N will lead to more operations of the RC approach and hard to find the optimization solution for the SVR algorithm. Fortunately, the RC approach can work well even if the N is less than 1000.

Fig. 4 illustrates the curves of the average cost function $J(\mathbf{w}_{out})$ of the RC approach with data length of $N = 500$, 1000, 1500 and 2000, respectively. It is obvious that the average of cost function keeps decreasing until the algorithm approaches a convergence. The iteration is only about 50 times even for 64QAM input case and the algorithm is always convergent. Meantime, from these curves, we can see that the convergence rate of the RC approach with shorter data length is faster than that with longer data length. Furthermore, it is implied that the cost function is independent of data length. It is worth mentioning that there is a sharp descent followed by a gradual ascent in the beginning of iteration, due to the effect of the transient process and magnification λ (see section 3.4), respectively.

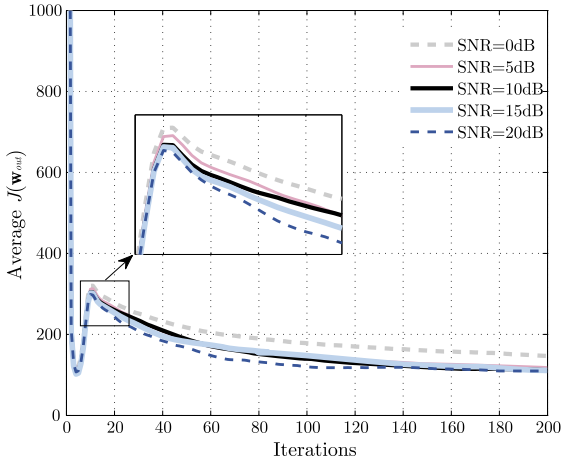

Fig. 5. 64QAM case, the average cost function $J(\mathbf{w}_{out})$ of the RC approach with SNR, $L_w = 9$, $N = 1000$.

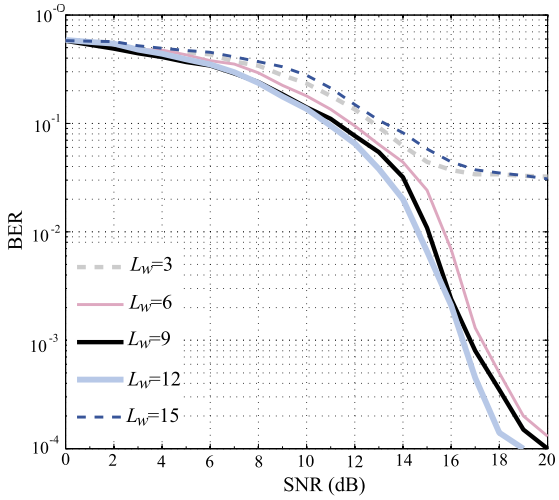

Fig. 6. 64QAM case, the average BER curves of the RC approach with several different length of readout weight L_w when length of data $N = 1000$.

From Fig. 5, we can see that the proposed RC approach is insensitive to noise, since the reservoir weight matrix is constructed by true signal subspace information. Fig. 6 illustrates the BER performance of the RC approach with several different lengths of readout weight L_w (= 3,6,9,12,15) for $N = 1000$. The approach appears to be quite robust in the sense that as long as the length of readout weight is moderate (e.g., $3 \leq L_w \leq 15$). It is obvious that too large value of L_w (e.g., $L_w = 15$) or too small L_w (e.g., $L_w = 3$), will cause signal detection failure. Fig. 7 shows the curves of the average cost function $J(\mathbf{w}_{out})$ of the RC approach with several different lengths of readout weight L_w. We can see that the RC approach is always convergent, no matter what the value of L_w.

The following experiments are based on signal-to-noise ratio (SNR) = 20 dB, $L_w = 9$, $N = 1000$, and all figures are drawn from one independent trial.

Fig. 8 shows the curves of the average cost function of the RC approach with linear and nonlinear readout functions when $L_w = 9$ and $N = 1000$. As we see from this figure, RC approach can work well no matter weather the readout function is linear or nonlinear. Meantime, it is clear that the case of nonlinear readout function shows more convergence precision. Fig. 9 illustrates the phase trajectories of three signal points using the RC approach with different attenuation factors a (= 2,5,10,20). For clarity, only 200 signal points are given out in these figures. The hollow circles denote the ideal 64QAM signal constellation points, all lines denote

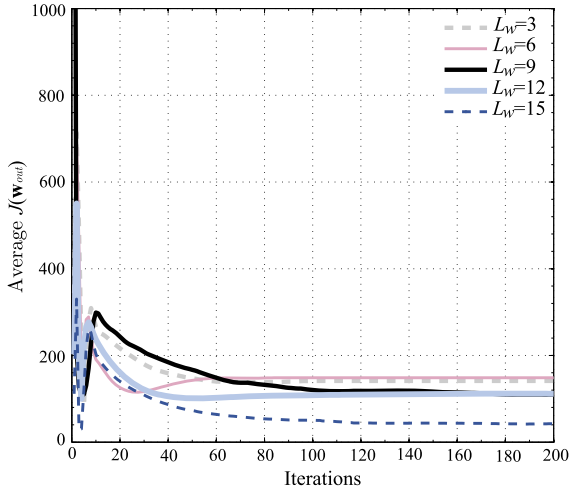

Fig. 7. 64QAM case, the curves of the average cost function $J(\mathbf{w}_{out})$ of the RC approach with several different lengths of readout weight L_w when $N = 1000$.

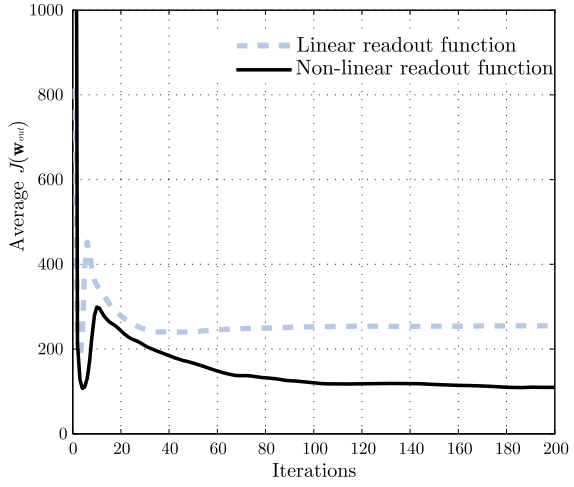

Fig. 8. 64QAM case, the curves of the average cost function $J(\mathbf{w}_{out})$ of the RC approach with linear and nonlinear readout functions when $L_w = 9$ and $N = 1000$.

the phase trajectories of signal points with the iteration, respectively, and the five solid lines express the trajectories of the given different signal points, respectively. In these figures, the left sub-figures are the *Transient Process*, the middle sub-figures the *Static Process*, and the right sub-figures the constellations of output signals of the RC approach, respectively. We can see that the phase trajectories are different and irregular with different attenuation factors a, but all of them will reach their respective true signal points when the algorithm is convergent, no matter what a may be. The factor a affects both the smooth degree and route of these trajectories.

5.2. The comparison with MMA and CMA algorithms

The results of comparison between RC method and the classical MMA, Constant Module Algorithm (CMA) [25] are shown in Fig. 10 and Fig. 11. Here, $N = 20000$ is adopted for MMA and CMA, but $N = 1000$ only for the RC approach, the step size for MMA and CMA is 1.8×10^{-8}. It is found that the RC approach is superior to those of the above mentioned BSD methods in performance. Furthermore, the data length of the RC approach is only 5%–10% of the MMA and CMA approach, and the iterations of the RC approach is 2.5% of the MMA and CMA approach, respectively. In addition, the convergence of RC is much faster than MMA and CMA.

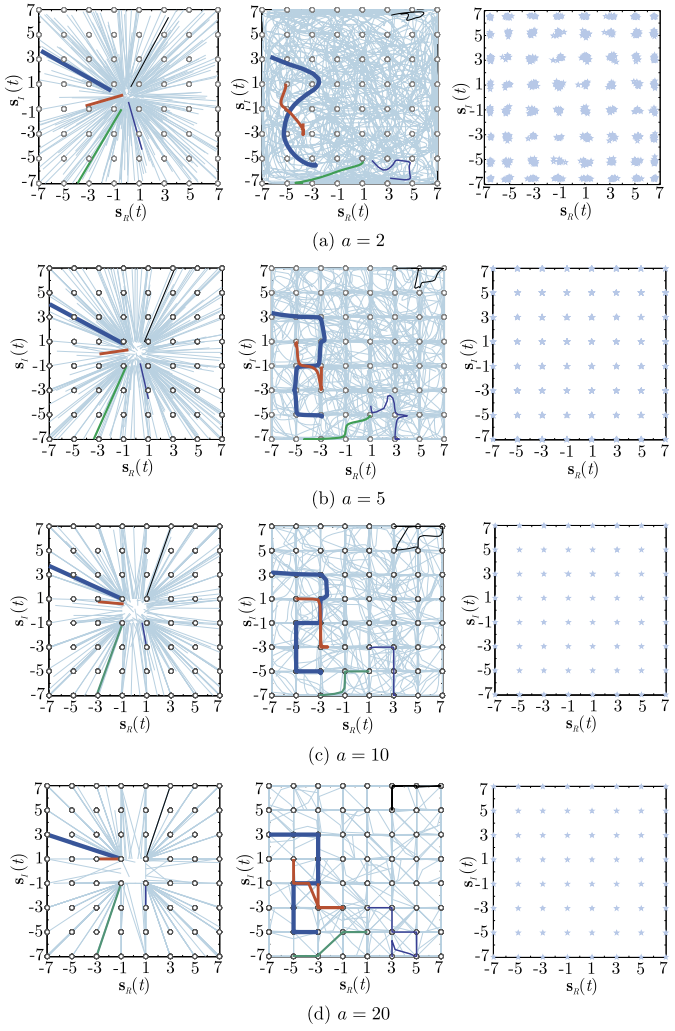

Fig. 9. The phase trajectories of the five given signal points using RC approach with different attenuation factors a.

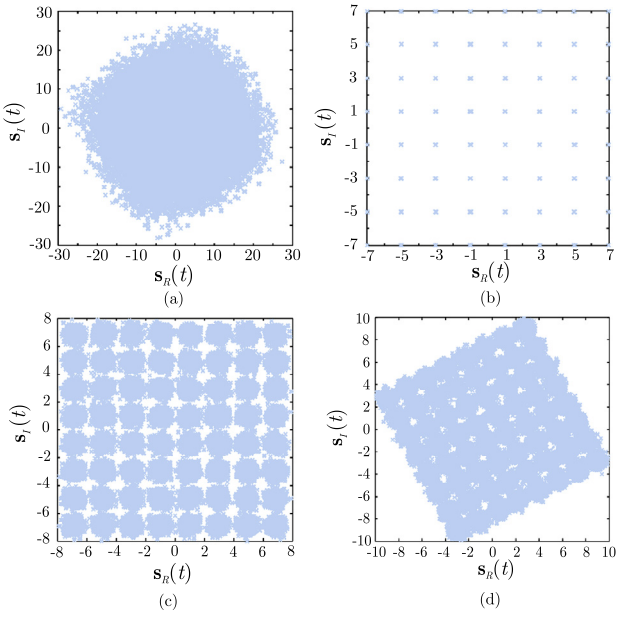

Fig. 10. 64QAM case, constellation diagrams of input and output signals for a single run: (a) the received signal, (b) output signals of the RC approach, (c) output signals of MMA approach, (d) output signals of CMA approach.

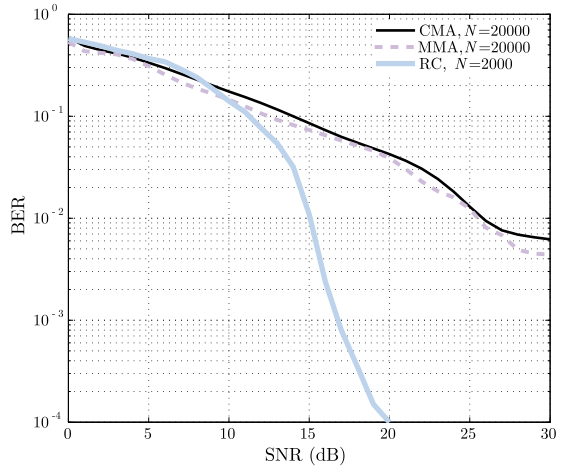

Fig. 11. 64QAM case, BER performance of RC, MMA and CMA as a function of SNR.

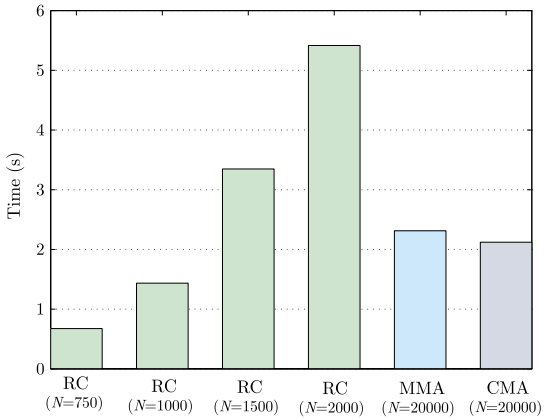

Fig. 12. 64QAM case, the running time of one independent trial of RC, MMA and CMA.

If we make a fair comparison to measure the training time, it turns out that our approach requires a longer training time than other algorithms in many cases. Fig. 12 illustrates the running time of one independent trial (Run in SNR = 30 dB and run one time) of the RC, MMA and CMA approaches. (CPU: Dual core Inter i5 1.7 GHz; Memory: 4G; Window 7 64 bit; Matlab2012a). Unfortunately, matrix operation is engaged in the RC approach. It is obvious that the running time of the RC approach (i.e., $N \leq 1000$) is less than that of MMA and CMA approaches. Meantime, the running time of the RC approach (i.e., $N = 1500, 2000$) is more than that of MMA and CMA approaches due to it requires a matrix inversion and needs to update diagonal matrices at each iteration. However, it is possible to reduce the running time by applying more efficient algorithms for matrix inversion and multiplication.

The curves of the average cost function $J(\mathbf{w}_{out})$ of the RC approach with blind and cooperative scenario have been shown by Fig. 13. It is clear that the convergence speed and precision of cooperative scenario are better than that of blind case, which will lead to a better performance.

Moreover, we will provide some results in the next section with numerical and simulation of comparison with Complex-valued Multilayer Perceptron (CMLP) [26] and SVM based on the Iterative Re-weighted Least Square (IRWLS) [27] methods.

5.3. The comparison with CMPL, and SVM methods

Fig. 14 shows the BER of RC, CMPL [26] and SVM [27]. CMLP works well with QAM signals as blind equalizers, and that they

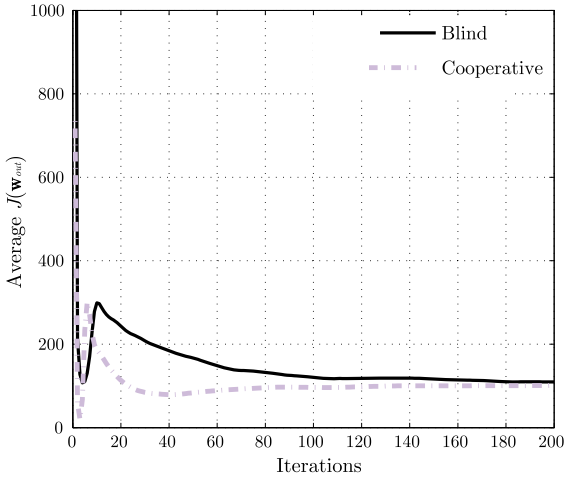

Fig. 13. 64QAM case, the curves of the average cost function $J(\mathbf{w}_{out})$ of the RC approach with blind and cooperative scenario when $L_w = 9$ and $N = 1000$.

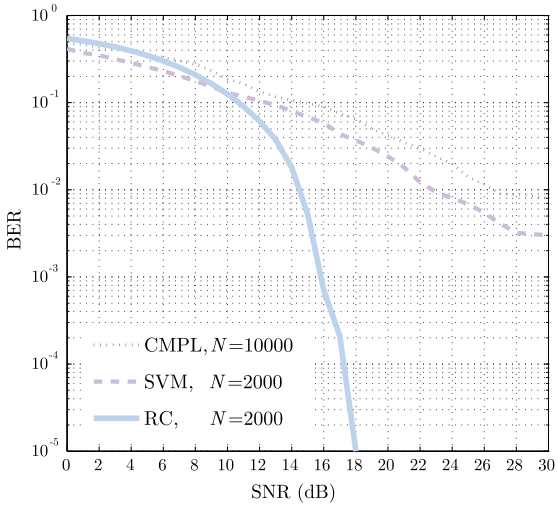

Fig. 14. 64QAM case, the BER of RC, CMPL and SVM.

can correct an arbitrary phase rotation introduced by the channel distortion even if there is no carrier phase tracking procedure [26]. However, a large data packet ($N = 20000$ for 64QAM) is needed for this method. The SVM method clearly provides a higher probability of convergence, achieving a full convergence with a relatively low number of samples. Unfortunately, its probability of correct convergence is only about 97 percents.

6. Conclusion

Unlike those existing BSD algorithms, the proposed approach can detect the source signal sequence directly without getting the equalizer coefficients and estimating channel impulse response at the receiver. The proposed approach guarantees a convergence within a short data packet and, therefore, can work in systems with a much shorter data record and faster time-varying channels.

As a new BSD approach, there are many interesting research topics in this area in the future. For instance, in this study we only used one random initial sequence to drive the RC network. It is feasible that a initial sequence can be designed from the view of signal space to accelerate convergence rate. Furthermore, examining how to optimally choose the parameters λ and a for general problems and their effect on the convergence will be another interesting research topic. We have fixed these parameters after only a few trials, which results in the excellent convergence

efficiency in all simulation cases in Section 6. In addition, training algorithms need to choose parameters experimentally for the best training results, for instance, SVR needs experimental selection of: soft margin, and the size of insensitive tube. How to select them intelligently is worthy to investigate. Meantime, the mathematical mechanism of the reservoir weight is under studying.

Acknowledgments

The authors would like to acknowledge the financial support of this work from the National Natural Science Foundation of China (NSFC) (grant nos. 61671329, 61201426, 61501331, 61303210), the Zhejiang Provincial Natural Science Foundation of China (NSFC) (grant nos. LQ16F010010, LY16F010016, LQ16F01001), Scientific Research Project of Education Department of Zhejiang Province of China (grant nos. Y201327231, Y201430529). The authors also would like to thank the reviewers for their valuable suggestion and insightful comments that helped in improving the quality of this manuscript.

Appendix A

A.1. The proof of Theorem 1

Proof. For all $\mathbf{Y} \in \mathbb{C}^{1\times N}$, it holds that

$$\mathbf{Y}\mathbf{W}_{res}\mathbf{Y}^H = \mathbf{Y}\mathbf{Q}\mathbf{Q}^H\mathbf{Y}^H = \mathbf{Y}\mathbf{Q}(\mathbf{Y}\mathbf{Q})^H \geq 0$$

thus $\mathbf{W}_{res}$ is a non-negative definite matrix. Since $\mathbf{W}_{res} = \mathbf{Q}\mathbf{Q}^H$, we get

$$\mathbf{W}_{res} = \mathbf{W}_{res}^H = \mathbf{W}_{res}^2$$

which implies $\mathbf{W}_{res}$ is an idempotent matrix.

By denoting λ as an eigenvalue of $\mathbf{W}_{res}$, and a nonzero vector $\mathbf{x} \in \mathbb{C}^{1\times N}$, then we have

$$\mathbf{W}_{res}\mathbf{x} = \lambda\mathbf{x}$$

and the idempotence of $\mathbf{W}_{res}$ implies that

$$\mathbf{W}_{res}\mathbf{x} = \mathbf{W}_{res}\mathbf{W}_{res}\mathbf{x} = \lambda\mathbf{W}_{res}\mathbf{x} = \lambda^2\mathbf{x}.$$

It follows that

$$\mathbf{x}^H\mathbf{W}_{res}\mathbf{x} = \lambda\mathbf{x}^H\mathbf{x} = \lambda^2\mathbf{x}^H\mathbf{x}$$

and thus

$$\lambda(1-\lambda)\mathbf{x}\mathbf{x} = 0.$$

So $\lambda = 0$ or $\lambda = 1$. Hence, the spectral radius of matrix $\rho(\mathbf{W}_{res}) = 1$. This proof is completed. □

Appendix B

B.1. The II term of (24) which be expressed in matrix notation

$$\begin{aligned} III &= \sum_{i=L_w-1}^{N} 2A_i y_i^* y_i \mathbf{u}_i^T \mathbf{w}_{out} \mathbf{u}_i^* \\ &= \sum_{i=L_w-1}^{N} \cdot \left\{ 2A_i |y_i|^2 \right. \\ &\quad \cdot \left\{ \begin{bmatrix} w_0 u_i^* u_i + w_1 u_i^* u_{i-1} + \cdots + w_{L_w-1} u_i^* u_{i-L_w+1} \\ w_0 u_{i-1}^* u_i + w_1 u_{i-1}^* u_{i-1} + \cdots + w_{L_w-1} u_{i-1}^* u_{i-L_w+1} \\ \vdots \\ w_0 u_{i-L_w+1}^* u_i + w_1 u_{i-L_w+1}^* u_{i-1} + \cdots + w_{L_w-1} u_{i-L_w+1}^* u_{i-L_w+1} \end{bmatrix} \right\} \\ &= \underbrace{\begin{bmatrix} u_{L_w-1}^* & u_{L_w}^* & \cdots & u_N^* \\ u_{L_w-2}^* & u_{L_w-1}^* & \cdots & u_{N-1}^* \\ \vdots & \vdots & \ddots & \vdots \\ u_0^* & u_1^* & \cdots & u_{N-L_w+1}^* \end{bmatrix}}_{U^H} \\ &\quad \cdot \underbrace{\begin{bmatrix} A_{L_w-1} & 0 & \cdots & 0 \\ 0 & A_{L_w} & \ddots & \vdots \\ \vdots & 0 & \ddots & \\ 0 & & \ddots & 0 \\ 0 & \cdots & 0 & A_N \end{bmatrix}}_{A} \\ &\quad \cdot \underbrace{\begin{bmatrix} |y_{L_w-1}|^2 & 0 & \cdots & 0 \\ 0 & |y_{L_w}|^2 & \ddots & \vdots \\ \vdots & 0 & \ddots & \\ 0 & & \ddots & 0 \\ 0 & \cdots & 0 & |y_N|^2 \end{bmatrix}}_{Y} \\ &\quad \cdot \underbrace{\begin{bmatrix} u_{L_w-1} & u_{L_w-2} & \cdots & u_0 \\ u_{L_w} & u_{L_w-1}^* & \ddots & u_1 \\ \vdots & \vdots & \ddots & \vdots \\ u_N & u_{N-1} & \cdots & u_{N-L_w+1} \end{bmatrix}}_{U} \cdot \underbrace{\begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_{L_w-1} \end{bmatrix}}_{\mathbf{w}_{out}} \end{aligned}$$

Appendix C. Supplementary material

Supplementary material related to this article can be found online at http://dx.doi.org/10.1016/j.dsp.2016.10.012.

References

[1] I. Fatadin, D. Ives, S.J. Savory, Blind equalization and carrier phase recovery in a 16-QAM optical coherent system, J. Lightwave Technol. 27 (17) (2009) 3042–3049.

[2] L. Hanzo, S.X. Ng, T. Keller, W. Webb, Quadrature Amplitude Modulation: From Basics to Adaptive Trellis-Coded, Turbo-Equalised and Space-Time Coded OFDM, CDMA and MC-CDMA Systems, Wiley–IEEE Press, New York, NY, USA, 2003.

[3] C. Wang, C. Lin, Q. Chen, B. Lu, X. Deng, J. Zhang, A 10-Gbit/s wireless communication link using 16-QAM modulation in 140-GHz band, IEEE Trans. Microw. Theory Tech. 61 (7) (2013) 2737–2746.

[4] C.Y. Chi, C.C. Feng, C.H. Chen, C.Y. Chen, Blind Equalization and System Identification: Batch Processing Algorithms, Performance and Application, Springer-Verlag, New York, NY, USA, 2006.

[5] H. Jaeger, W. Maass, J. Principe, Special issue on echo state networks and liquid state machines, Neural Netw. 20 (3) (2007) 287–289.

[6] H. Jaeger, The "echo state" approach to analysing and training recurrent neural networks, GMD-German National Research Institute for Computer Science, 2001, Tech. Rep. 148, 1–48 pp.

[7] W. Maass, T. Natschläger, H. Markram, Real-time computing without stable state: a new framework for neural computation based on perturbations, Neural Comput. 14 (11) (2002) 2531–2560.
[8] T. Hastie, R. Tibshirani, J. Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 2nd ed., Springer-Verlag, New York, NY, USA, 2008.
[9] A.J. Smola, B. Schölkopf, A tutorial on support vector regression, Stat. Comput. 14 (2004) 199–222.
[10] S. Salcedo-Sanz, J.-L. Rojo-Álvarez, M. Martínez-Ramón, G. Camps-Valls, Support vector machines in engineering: an overview, WIREs Data Min. Knowl. Discov. 4 (3) (2014) 234–267.
[11] Digital transmission standard for cable television, American National Standard (ANSI/SCTE 07 2006), 2006, pp. 1–20.
[12] G. Manjuath, H. Jeager, Echo state property linked to an input: exploring fundamental characteristic of recurrent neural networks, Neural Comput. 25 (3) (2013) 671–696.
[13] M. Lukoševičius, H. Jeager, Reservoir computing approaches to recurrent neural network training, Comput. Sci. Rev. 3 (3) (2009) 127–149.
[14] X. Dutoit, B. Schrauwen, J.V. Campenhout, D. Stroobandt, H.V. Brussel, M. Nuttin, Pruning and regularisation in reservoir computing: a first insight, in: Proc. European Symposium on Artificial Neural Networks (ESANN), Bruges, Belgium, Apr. 2008, pp. 1–6.
[15] X. Li, H. Fan, QR factorization based blind channel identification and equalization with second-order statistics, IEEE Trans. Signal Process. 48 (1) (2000) 60–69.
[16] G. Golub, C. Loan, Matrix Computations, 2nd ed., Johns Hopkins Univ. Press, Baltimore, MD, 1996.
[17] L. Mantas, H. Jaeger, B. Schrauwen, Reservoir computing trends, Künstl. Intell. 26 (4) (2012) 365–371.
[18] F. Triefenbach, J.P. Martens, Can non-linear readout nodes enhance the performance of reservoir-based speech recognizers, in: Proc. on Int. Conf. Informatics and Computational Intelligence, Bandung, Dec. 2011, pp. 262–267.
[19] X. Ruan, Y. Zhang, Blind sequence estimation of MPSK signals using dynamically driven recurrent neural networks, Neurocomputing 129 (2014) 421–427.
[20] C.J. Wild, G.A.F. Seber, Chance Encounters: A First Course in Data Analysis and Inference, Wiley, New York, NY, USA, 2000.
[21] A.H. Sayed, Fundamentals of Adaptive Filtering, Wiley–IEEE Press, New York, NY, USA, 2003.
[22] S. Abrar, A.K. Nanid, Blind equalization of square-QAM signals: a multimodulus approach, IEEE Trans. Commun. 58 (6) (2010) 1674–1685.
[23] O. Shalvi, E. Weinstein, Super-exponetial methods for blind deconvolution, IEEE Trans. Inf. Theory 39 (2) (1993) 504–519.
[24] J. Shen, Z. Ding, Direct blind MMSE channel equalization based on second-order statistics, IEEE Trans. Signal Process. 48 (4) (2000) 1015–1022.
[25] R. Johnson, P. Schniter, T.J. Endres, J.D. Behm, Blind equalization using the constant modulus criterion: a review, Proc. IEEE 86 (10) (1998) 1927–1950.
[26] C. You, D. Hong, Nonlinear blind equalization schemes using complex-valued multilayer feedforward neural networks, IEEE Trans. Neural Netw. 9 (6) (1998) 1442–1455.
[27] M. Lazáro, J. González-Olásola, Blind equalization using the IRWLS formulation of the support vector machine, Signal Process. 89 (2009) 1436–1445.

Xiukai Ruan received his M.Sc and Ph.D. degrees from Nanjing University of Posts and Telecommunications, Nanjing, China, in 2006 and 2011, respectively. He is a commissioner of the Chinese Society of Astronautics and the Chinese Society for Optical Engineering. He is currently working as the chief engineer and vice director of the National-Local Joint Engineering Laboratory for Digitalize Electrical Design Technology. His research interests are broadly in the areas of signal and information processing of optical communication systems, countermeasure of optical transmission injury, and photoelectric detection automatic machine manufacturing technology.

Chang Li received his B.E. degree from the Zhejiang Normal University, Jinghua, China, in 1997, the M.Sc degree from Nanjing University of Posts and Telecommunications, Nanjing, China, in 2003. He is currently a Lecturer with the College of Physics and Electronic Information Engineering, Wenzhou University, China. His areas of special interest include signal processing and computational simulation for communication systems.

Weibo Yang received his B.E. and M.Sc degrees from Inner Mongolia University Of Science & Technology, Baotou, China, in 2000 and 2003, respectively. He is currently a Lecturer with the College of Physics and Electronic Information Engineering, Wenzhou University, China. His areas of special interest include embedded system and Artificial intelligence.

Guihua Cui received the B.S. and M.S. degrees in optical engineering from Beijing Institute of Technology, China, in 1984 and 1987, respectively, and Ph.D. degree in color science from University of Derby, UK, in 2000. He is currently a Professor with the College of Physics and Electronic Information Engineering, Wenzhou University, China. His research interests include uniform color space and color image processing.

Haiyong Zhu received the B.S. degree in optoelectronics from China Jiliang University in 2004, M.S. degree and Ph.D. degree in condensed matter physics from Fujian Institute of Research on the Structure of Matter, Chinese Academy of Science, in 2007 and 2010. His research has been concerned with laser technology and application.

Zhili Zhou received the M.Sc from Zhejiang University of Technolegy, Zhejiang, China, in 2009. He is currently pursuing his Ph.D. degree in Sun Yat-sen University. His research interests are broadly in the areas of signal and information processing of optical communication systems.

Yuxing Dai received his Ph.D. degree in Control Theories and Control Engineering from Central South University, China, in 2003. He is currently Professor at Wenzhou University, China. His research interests include power electronics, computer numerically controlled machine tools, and mircogrid.

Xiaojing Shi received her B.E. degree from the East China Normal University, Shanghai, China, in 1997, the M.E. and Ph.D. degrees in circuits and systems from University of Miyazaki, Japan, in 2002. She is currently a Lecturer with the College of Physics and Electronic Information Engineering, Wenzhou University, China. Her areas of special interest include signal processing and computational neuroscience.