Chapter

The Novel Applications of Deep Reservoir Computing in Cyber-Security and Wireless Communication

Kian Hamedani, Zhou Zhou, Kangjun Bai and Lingjia Liu

Abstract

This chapter introduces the novel applications of deep reservoir computing (RC) systems in cyber-security and wireless communication. The RC systems are a new class of recurrent neural networks (RNNs). Traditional RNNs are very challenging to train due to vanishing/exploding gradients. However, the RC systems are easier to train and have shown similar or even better performances compared with traditional RNNs. It is very essential to study the spatio-temporal correlations in cyber-security and wireless communication domains. Therefore, RC models are good choices to explore the spatio-temporal correlations. In this chapter, we explore the applications and performance of delayed feedback reservoirs (DFRs), and echo state networks (ESNs) in the cyber-security of smart grids and symbol detection in MIMO-OFDM systems, respectively. DFRs and ESNs are two different types of RC models. We also introduce the spiking structure of DFRs as spiking artificial neural networks are more energy efficient and biologically plausible as well.

Keywords: recurrent neural networks, reservoir computing, delayed feedback reservoir, echo state networks, cyber-security, smart grids, MIMO-OFDM

1. Introduction

Smart grids are a new generation of power grids, which provide more intelligent and efficient power transmission and distribution. However, the smart grids are vulnerable to security challenges unless properly protected. False data injection (FDI) attacks are the first and most common type of attacks in smart grids. Two major types of FDI attacks are known in smart grids. These two major types are single-period or opportunistic and multi-period or dynamic attack, respectively. In single-period attack, the adversary waits until it finds the opportunity to launch the attack instantaneously. On the other hand, in dynamic attacks, the adversary launches the attack gradually and through time toward its desired state. The single-period attacks are widely studied in the literature and they are more easily detected by the supervisory control and data acquisition (SCADA). In this chapter, we focus to study the multi-period or dynamic attacks [1–5].

State vector estimation (SVE) is the first technique to tackle the FDI detection in smart grids. However, SVE fails to detect stealth FDI attacks with low magnitudes.

In recent years, both supervised and unsupervised machine learning (ML) approaches have been proposed to study FDI detection in smart grids. Generally, ML-based techniques have shown better performances than SVE. However, the ML techniques that have been proposed so far are not capable to capture the rich spatio-temporal correlations that exist between different components of smart grids. Therefore, in this chapter, we introduce spiking delayed feedback reservoirs (DFRs) to tackle the FDI detection problem in smart grids as they are very energy efficient and also can capture the spatio-temporal correlations between different components of smart grids. DFRs are an energy efficient class of reservoir computing systems [6–8].

Figure 1 demonstrates the structure of a reservoir computing (RC) system. As it can be seen, there are three layers in RC systems. They are the input, reservoir, and output layer, respectively. The architecture of RC systems is based on recurrent neural networks (RNNs). However, unlike the RNNs, the weights of the hidden (reservoir) layer are fixed and do not go through a training. The reservoir weights have to be initialized such that the echo state property is satisfied. Echo state property implies that in order to form a memory, the largest eigenvalue of the reservoir weights has to be less than 1. The largest eigenvalue of the reservoir layer's weights is a design parameter and plays an important role in the performance of the RC systems. DFRs, echo state networks (ESNs), and liquid state machines (LSMs) are three different categories of RC systems. The strength of RNNs is employed as the reservoir or liquid states. In the reservoirs or liquid states, the weights of synaptic connections are fixed and do not require any training. The output weights are the only sets of weights that require training in RC models. This results in reducing the computational complexity of RC models compared to traditional RNNs [9–12].

Equation (1) expresses the states of reservoir nodes,

$$s(t) = f\left[W_{res}^{res}.s(t-1) + W_{in}^{res}.x(t-1)\right], \qquad (1)$$

where $s(t)$ is the state of reservoir node at time t; $x(t-1)$ corresponds to the input signal at time $t-1$; W_{res}^{res} and W_{in}^{res} correspond to the weights of randomly generated reservoir and input connection, respectively; and $\hat{y}$ represents the estimated output that can be expressed in terms of input and weight connections,

$$\hat{y} = W_{res}^{out}s(t) + W_{in}^{out}.x(t-1) + W_{bias}^{out}, \qquad (2)$$

where W_{res}^{out} are the output weights of the neurons that form the reservoir layer; W_{in}^{out} correspond to the feedback weights from output layer to reservoir layer; and

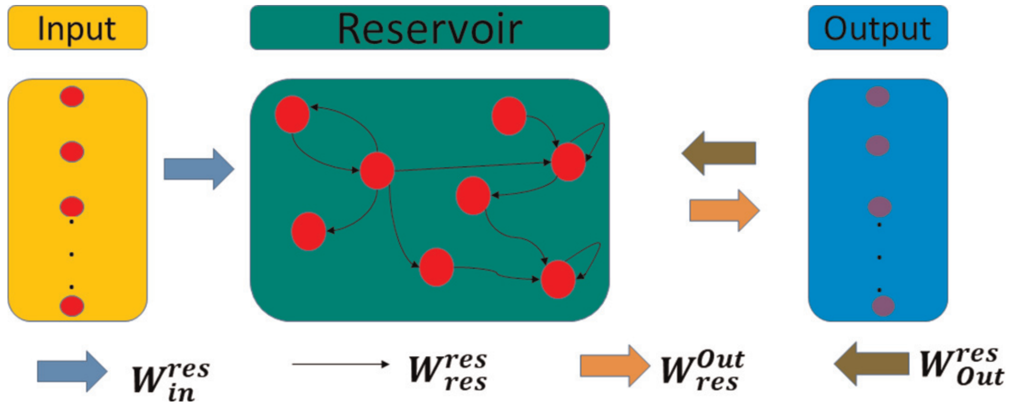

Figure 1.
Structure of reservoir computing.

W_{bias}^{out} is the set of weights for bias values training. The process of nonlinear mapping is accomplished by the neurons in the reservoir layer. The neurons in the reservoir layer own two major properties: (1) high dimensionality and (2) forming a short term memory that spatio-temporal patterns can be memorized. Several studies have shown that these two properties are satisfied only if the neurons at the reservoir layer operate at the edge of chaos. Satisfying the echo states property, is the key to make the reservoir neurons work at the edge of chaos. The lower computational complexity and the flexible reservoir implementation of RC models make them very suitable for unconventional computing paradigms applications.

The DFR is a ring topology of RC systems, where a single artificial neuron and a delay loop together form the reservoir layer. There are multiple choices available for the single artificial neuron of the DFR. In this chapter, we introduce spiking neurons as the nonlinear single neuron of the DFR. Spiking neurons are one of the several mathematical models that are introduced to model the biological neurons. Spikes are the main signals that the neurons of the brain use for communication. Hence, the mathematical representation of the biological neurons as spikes tends to be more biological plausible. Energy efficiency is another motivation to use the spiking neurons. TrueNorth chip consumes only 70 milliWatts (mW) to run 1 million spiking neurons with 256 million synapses [13–15]. The energy efficiency of spiking neural networks (SNNs) makes them a suitable choice for hardware implementations of artificial neurons as well [16, 17].

So far, several models for spiking neurons including leaky-integrate-and-fire (LIF) and the Hodgkin-Huxley have been proposed to mimic the behavior of our brains' neurons [18]. The LIF models of spiking neurons have been used more commonly than other spiking artificial models of neurons due to their simplicity and ease of hardware implementation [19, 20]. The spiking neurons fire a spike as soon as a stimulating current is applied on their membrane, which makes the voltage of the membrane exceeds a certain threshold value. The relationship between the stimulating current and the voltage of membrane is expressed as follows:

$$\tau_m \frac{dv_m}{dt} = -(V_m - E) + (I_{noise} + I_s)R_m, \tag{3}$$

where V_m is the membrane voltage; $\tau_m = R_m C_m$ corresponds to the neuron's time constant; C_m and R_m are the capacitance and the resistance of the membrane, respectively; E represents the resting voltage; I_{noise} is noise current; and I_s is stimulus current [21]. We set R_m to 1 mega *ohms* and C_m = 10 nano Farads (nF).

In **Figure 2**, the topology of our proposed spiking DFR is demonstrated. There are multiple blocks in this structure. The input block is where the smart grids' measurements are received. These measurements have to be first encoded before getting processed by DFR. There are two major types of encoding schemes for spiking neurons, namely rate encoding and temporal encoding [22]. Rate encoding has been vastly studied in the literature. However, recent studies have shown that temporal encoding schemes are more efficient and are superior to rate encoding schemes. The exact time that spike fires is used for temporal encoding of spikes. However, in rate encoding schemes, the number of the spikes that are fired by the neuron is used to encode the stimulus.

It has been shown in several experiments that temporal encoding is more likely to be the encoding scheme, which is leveraged by biological neurons. The neurons in the lateral geniculate nucleus, retina, and the visual cortex respond to the stimuli with milliseconds (ms) precision. The computational complexity of temporal

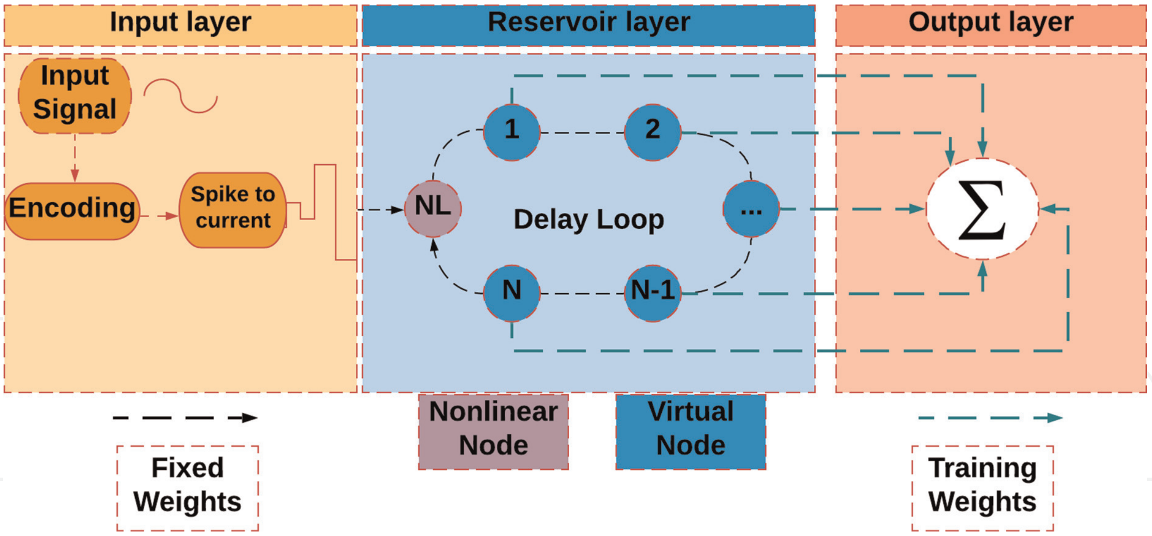

Figure 2.
Spiking delayed feedback reservoir computing.

encoding schemes has also made them superior to rate encoding approaches [23]. Therefore, in this chapter, we focus on temporal encoding schemes.

After the smart grids' measurements are encoded, the encoded data is then converted to the analog current. This current is next fed in to the nonlinear node, which in our case, is a LIF neuron. For each current signal, its corresponding spike train is generated by the LIF neuron, and this spike train goes through a delay loop. The delay loop along with the LIF neuron forms the reservoir layer of DFR. We repeat this process as long as the corresponding reservoir states of each smart grid's measurements are generated. The interspike intervals (ISI) of each spike trains are used as the training feature of the readout layer [24]. In this chapter, a multi-layer perceptron (MLP) is used as the readout layer. The features extracted in the reservoir layer are used for training the MLP layer. For each class of data, i.e., compromised and uncompromised, a proper label is assigned. We consider 1 as the label of compromised samples, and 0 for uncompromised samples.

Equation (4) expresses the governing equation for DFR,

$$\dot{x} = -x(t) + F(x(t-\tau), I(t), \theta), \tag{4}$$

where F is a differentiable nonlinear function; τ is the delay loop, which is a hyperparameter that requires tuning; $x(t)$ corresponds to the reservoirs states of DFR; and $I(t)$ is the input stimulus current signal along with a masking scheme. The total delay time, τ, is divided into N equidistant delay units within the delay loop. Dividing the total delay into N equidistant delay units is expressed as follows:

$$\tau = N\theta, \tag{5}$$

where θ represents the time interval between reservoir virtual nodes. Unlike the conventional RC model, the number of nonlinear nodes of DFR is drastically reduced, due to the ring topology of DFR. The weights of the output MLP layer are the only weights that undergo the training process [16].

DFRs have drawn a lot of attentions due to their capability to map the data from low dimensional space to high dimensional space. As it can be seen in **Figure 3**, by mapping the data from low dimensional space to high dimensional space, the nonlinearly separable data becomes linearly separable. The chaos theory through Lyapunov analysis has shown that delay systems can show high dimensional behavior if the delay value is tuned properly such that the delay system operates at the edge of chaos. The Lyapunov dimension of a delay chaos system directly is

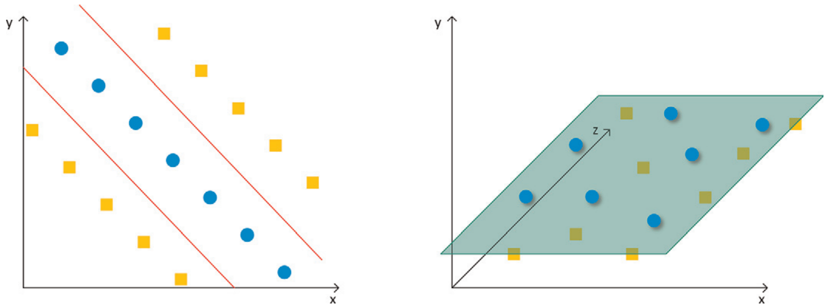

Figure 3.
High dimensional mapping of data using DFR.

determined by to the delay value [25]. In this chapter, we will examine the effect of delay value on the performance of DFR while detecting the dynamic hidden attacks in smart grids.

In this chapter, we will also look at symbol detection in multiple-input multiple-output orthogonal frequency division multiplexing (MIMO-OFDM) systems. In wireless communication systems, multicarrier access techniques are realized through OFDM. In fact, frequency-selective fading channels are converted to multiple flat-fading subchannels [26–28]. Spectral efficiency, transceiver structure, channel capacity, and robustness against interference are all improved as a result of applying OFDM in wireless communication systems [29–33]. MIMO systems are also extensively leveraged in different wireless communication systems including HSPA+(3G), WiMAx(4G), and long term evolution (4G LTE). By using MIMO systems, the capacity of wireless link is improved through the transmission of symbols on multiple paths. The system which is realized through the combination of MIMO and OFDM systems is called a MIMO-OFDM system [34–38]. A MIMO-OFDM system has shown to be very effective in utilizing the benefits of both MIMO and OFDM systems.

In order to detect the transmitted symbols accurately at the receiver (Rx), it is very essential to estimate the wireless channel state information (CSI) precisely [39–41]. CSI estimation is one of the major challenges of MIMO-OFDM systems. There are generally two major approaches for CSI estimation. The first approach leverages blind channel estimation to obtain the statistical properties of the channel [42]. The second category of CSI estimation techniques is based on training the symbols sent by transmitter (Tx) and received by (Rx) [29, 43, 44]. Training-based CSI estimation techniques have been adopted in many advanced communication systems including 3GPP LTE/LTE-Advanced. In the former category of CSI estimation techniques, no computational overhead is inferred, but they are good only for the channels that are varying very slowly with respect to time [45]. The latter category, i.e., training-based category can be applied for any channel regardless of their statistical properties. Therefore, the learning-based techniques including artificial neural networks have been vastly studied in literature [46–48] as the wireless channel estimation mechanism. RNNs have also been studied in [49–52] for CSI estimation and symbol detection. Due to the difficulties of training, the conventional RNNs, we introduce echo state networks (ESN) for symbol detection and CSI estimation in MIMO-OFDM wireless communication systems.

2. Problem formulation of smart grids attack detection

The state and topology of smart grids are the two major targets that are manipulated by the adversaries [53]. The state of the smart grids is the key factor in determining the measurements values. A linear function H and the environment noise are the other two factors that determine the measurements values.

$$z = Hx + n, \tag{6}$$

where z is the measurement vector that represents the real parts of the line flows and bus injections; H is a linear function; x is the state vector, and n is the environment noise [53]. Equation (6) can be written as follows in case the meters are compromised by an adversary,

$$\begin{aligned} \check{z} &= z + a, \\ \check{z} &= Hx + n + a, \end{aligned} \tag{7}$$

where a is the attack vector. The attack represented in Eq. (7) is an observable attack. The attack can also be hidden by the attacker. In this chapter, we consider the attacks as hidden dynamic attacks. The hidden attack is defined as $a = Hc$, and Eq. (6) is reformulated as follows,

$$\begin{aligned} \check{z} &= Hx + n + Hc \\ \check{z} &= H(x + c) + n, \end{aligned} \tag{8}$$

where c is the desired state of the adversary, where the attacker wants to drift the normal state of the smart grid toward its desired state by hiding it in the H matrix. Hidden attacks are more challenging to be detected. The adversaries launch dynamic attacks such that the state of the smart grid system is drifted toward their desired state gradually. Dynamic attacks are defined as a function of time as the adversary achieves its desired state gradually and through time. In single-period attacks, the variations of the attacks magnitude are sudden and abrupt, and are more easily detected. The formulation of dynamic attack used in this chapter is as follows:

$$\check{z}(t) = Hx(t) + n + a(t). \tag{9}$$

The dynamic attack $a(t)$ is time dependent, and we also assume that the adversary has access to H matrix. Thus, the attack can be performed as hidden or unobservable. In hidden attacks, the attack $a(t)$ can be expressed as $a(t) = Hc(t)$, and $c(t)$ is defined as follows:

$$c(t) = Acos\left(2\pi f_c t\right) \times N(0, 1), \tag{10}$$

where A is the magnitude of attack; cos is cosine function; f_c corresponds to the frequency of attack and we set that equal to 1 in this chapter, and $N(0,1)$ is a normally distributed vector in which its mean is zero and its variance is 1.

$$\check{z}(t) = H\left(x + Acos\left(2\pi f_c t\right) \times N(0, 1)\right) + n. \tag{11}$$

MATPOWER is a publicly available toolbox [54] that can be used to simulate the smart grids. In this chapter, we use MATPOWER to simulate the meters of a smart grid with 14 buses. There are totally 34 different meters in an IEEE-14 bus smart grid. We assume that the level of the access that the adversary can have to the meters of the system can range from 0 to 34. The level of access is defined as the number of meters that can be compromised by the attacker. In this chapter, the dataset that we use for train, test, and validation is assumed to be unbalanced.

A dataset is called unbalanced when the ratio of compromised and uncompromised samples is not equal. In this chapter, it is assumed that 80% of the samples are uncompromised and 20% are compromised. Totally, 10,000 samples for training and 10,000 samples for test and validation are generated using MATPOWER.

3. Attack detection performance of DFR

The performance metrics for evaluation are ***accuracy*** and ***F1***. ***Accuracy*** and ***F1*** are defined as:

$$\text{Accuracy} = (\text{TP} + \text{TN})/(\text{TP} + \text{TN} + \text{FP} + \text{FN}), \tag{12}$$

$$\text{F1} = 2 \times \frac{\textit{Precision} \times \textit{Recall}}{\textit{Precision} + \textit{Recall}}, \tag{13}$$

where $Precision = \frac{TP}{TP+FP}$; and $Recall = \frac{TP}{TP+FN}$ and TP, TN, FP, and FN correspond to the number of true positive, true negative, false positive, and false negative samples, respectively.

Accuracy of attack detection for three different methods and magnitude of attacks, *A* = 0.1, 1, and 10.

In order to evaluate the performance of our proposed spiking DFR model, we compare our results with a MLP and a SNN. The MLP is trained using backpropagation algorithm and SNN is trained using precise spike driven (PSD) algorithm. In PSD, temporal encoding is leveraged as the encoding scheme. PSD is used to learn the hetero-associations that exist in spatio-temporal spike patterns and is introduced in [21]. As it can be seen in **Figures 4** and **5**, spiking DFR + MLP outperforms both MLP and SNN in terms of ***accuracy*** and ***F1***. That is due to the fact that the spiking DFR + MLP is capable to map the data from low dimensional space to high dimensional space, and also captures the spatio-temporal correlation that exists between different components of smart grids. Based on our simulation results, the average **accuracy** of attack detection is increased up to **94.6%** when the combination of spiking neurons, DFR, and MLP is realized in a single platform. This improvement is observed for all different magnitude of attacks and number of compromised measurements. In our baseline model where only SNNs are used, the average **accuracy** is **77.92%**. This improvement implies that the average **accuracy** is improved about **17%** through our introduced hybrid spiking DFR and MLP model. ***F1*** measure shows even more significant improvement brought about. ***F1*** that is achieved through combination of spiking neurons, DFR, and MLP is **78%**. However, the ***F1*** which is achieved by SNN and PSD algorithm for dynamic attack detection is about **25%**, which means that our introduced model increases the ***F1*** for **53%**.

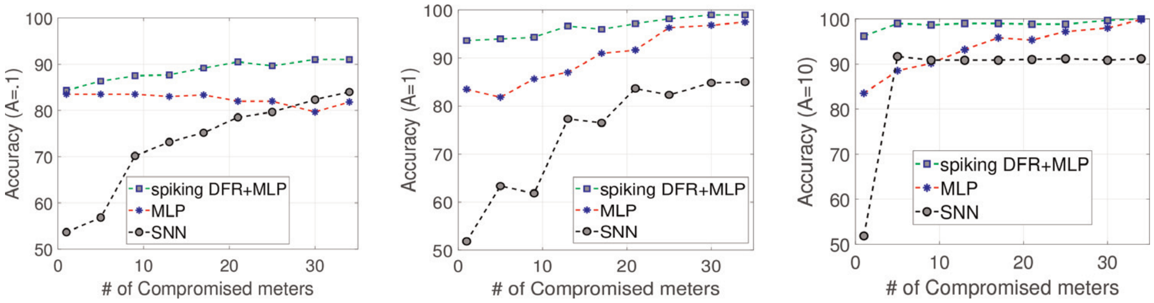

Figure 4.
Accuracy of attack detection for three different methods and magnitude of attacks, A = 0.1, 1, and 10.

3.1 Delay effect on the performance of DFR

As it was mentioned in Section 1, the DFRs cannot show high dimensional behavior unless the delay value is tuned properly that the DFR operates at the edge of chaos. At this part, we show that delay value can significantly affect the performance of DFR for hidden dynamic attack detection on smart grids. **Figure 6** demonstrates the performance of DFR for different values of delay. As it can be seen in **Figure 6**, for delay equal to 40 milliseconds (ms), the performance of spiking DFR + MLP achieves the highest value in terms of ***F1*** and ***accuracy***. However, for delay value equal to 10 ms, the lowest performances are obtained. This observation implies that only for a proper delay value, the spiking DFR + MLP can operate at the edge of chaos and show high dimensional behavior. The phase portrait behavior of DFR with respect to varying the delay time is shown in **Figure 7**. The dynamic behavior of the delay systems can be tracked through phase portraits and chaotic or periodic behavior of the system can be demonstrated. It is suggested in [25] that if the delay of dynamic system is tuned properly, it can show high dimensional behavior. We also investigate the solution of the delay differential equation (DDE) to further explore the dynamic behaviors of our introduced model. As demonstrated in **Figure 7**, DDE is leveraged to model the dynamic behavior of nonlinear function while the delay is varying.

Figure 7 shows that varying the delay value can shift the behavior of delay system from periodic to edge of chaos region and completely chaotic.

3.2 Complexity analysis

In this section, the complexity of our approach in terms of training time is analyzed. The computational complexity of the introduced spiking DFR + MLP

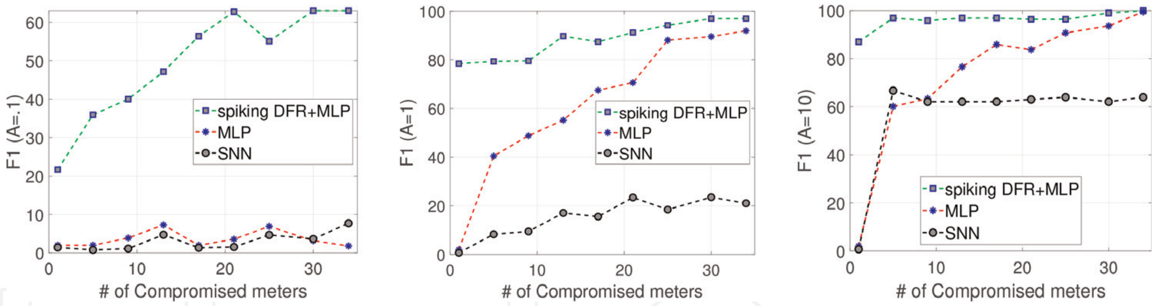

Figure 5.
F1 of attack detection for three different methods and magnitude of attacks, A = 0.1, 1, and 10.

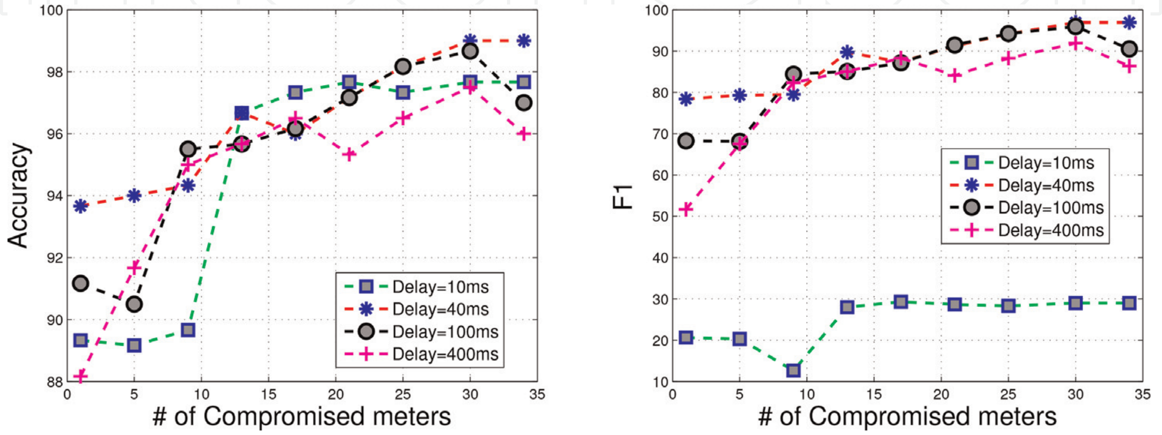

Figure 6.
Effects of different values of the delay on the performance when the A = 1.

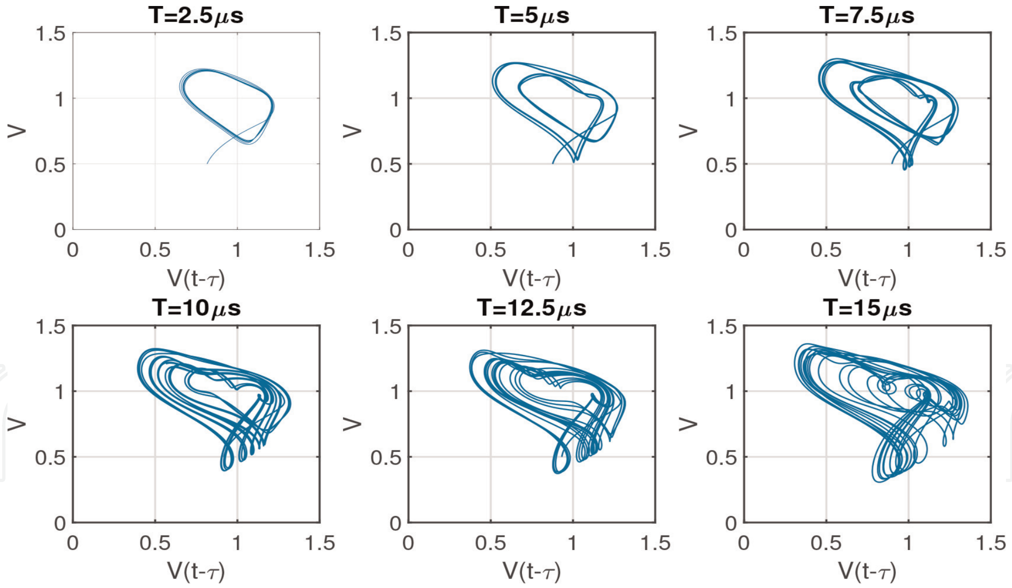

Figure 7.
Phase portraits of DFR.

Algorithm	Training time
Spiking DFR + MLP	16.69 s
MLP	3.2 s
SNN	90 s

Table 1.
Computational complexity analysis.

is associated with calculating the state of the reservoir layer, and updating the weights of readout layer during training. In the introduced spiking DFR model, the weights of input and reservoir layers are fixed and do not undergo any training. That is the fact that makes DFRs significantly computationally efficient compared to other types of RNNs. In traditional RNNs, all the hidden layers require to be trained. Due to the training of all hidden layers, the RNNs are very difficult to train. The measure of complexity is equivalent to the total number of floating-point operations (FLOPs). The training time of RC-based learning techniques correspond to the complexity of model as well [55]. In order to evaluate the computational complexity of our proposed model, the training time of our model is compared with the baseline approaches, i.e., MLP and SNN. **Table 1** presents the training times (complexity) of spiking DFR + MLP, MLP, and SNN.

The SNN which is trained by PSD algorithm shows the highest computational complexity, as it can be seen in **Table 1**. The spiking DFR + MLP and MLP rank as the second and third computationally complex algorithms, respectively. As it can be seen in **Figure 2**, there are some building blocks in the spiking DFR + MLP. Therefore, the computational complexity of spiking DFR + MLP is higher than a simple MLP. Temporal encoding, spike to current, and reservoir blocks are the blocks that exist in our introduced model. However, the superiority of our model in terms of performance makes it justified for us to use this model as the attack detection platform in smart grids.

4. Reservoir computing-based symbol detection

4.1 Received signal

We assume there are N_r antennas at Rx; and N_t antennas at Tx. The received signal can be expressed as:

$$\boldsymbol{y}(t) = \sum_i \boldsymbol{h}_i(t) \circledast x_i(t) + \boldsymbol{n}(t) \tag{14}$$

where $\boldsymbol{n}(t)$ is the additive noise; $\circledast$ stands for the convolution operation; $\boldsymbol{h}_i(t) \in \mathbb{C}^{N_r \times 1}$ is the channel from the ith Tx antenna to the Rx; and $x_i(t)$ is the associated transmitted signal, which is defined as:

$$x_i(t) = \sum_{p=0}^{\infty} \sum_{n=0}^{N_c} g\left(t - pT_s\right) s_i[n,p] e^{j\pi\left(nf_0 + f_c\right)t} \tag{15}$$

where n is the index of subcarrier; p is the index of time instance; f_c is the carrier frequency; $s[n,p]$ is modulation symbols; f_0 is the frequency space between each subcarrier component; N_c is the number of subcarriers; and $g(t)$ is the waveform function with finite time support which is usually selected as:

$$g(t) = \begin{cases} 1 & \mathrm{t} \in (0, T_s] \\ 0 & \text{otherwise} \end{cases}$$

The channel model is defined according to the ray-tracing principle

$$\boldsymbol{h}(t) := \sum_k \alpha_k \boldsymbol{a}(\theta_k) \delta(t - \tau_k) \tag{16}$$

where k is the index of channel taps; θ_k stands for the angle of arrival (DoA); α_k is the associated path gain; and τ_k is the delay parameter.

4.2 Symbol detection framework

In symbol detection, we aim to estimate $s[n,p]$ belonging to all transmission antennas and time channel use, where the general framework is shown in **Figure 8**. For this problem, the interference from different antennas and OFDM symbols need to be canceled out. Rather than estimating the underlying channel information, in our approach, the reservoir computing network $\mathcal{RC}$ is applied to $\boldsymbol{y}(t)$ to retrieve the transmitted waveform. At the learning stage, the objective is written as:

$$\min_{\mathbf{W}_{out}} L\left(\mathcal{RC}\left(\boldsymbol{y}(t)\right), \{x_i(t)\}_i\right) \tag{17}$$

where L is the loss function. Through learning the output weight of RC, it yields an interference cancellation manner, which can recover the transmitted signals. Meanwhile, this relies on a symbol level synchronization among multiple antennas. Alternatively, the symbol detection can be learned through a decomposed manner.

Following this way, we can rewrite the received signal model (14) as:

$$\boldsymbol{y}(t) = \boldsymbol{h}_k(t) \circledast x_k(t) + \sum_{j \neq k} \boldsymbol{h}_j(t) \circledast x_j(t) + \boldsymbol{n}(t) \tag{18}$$

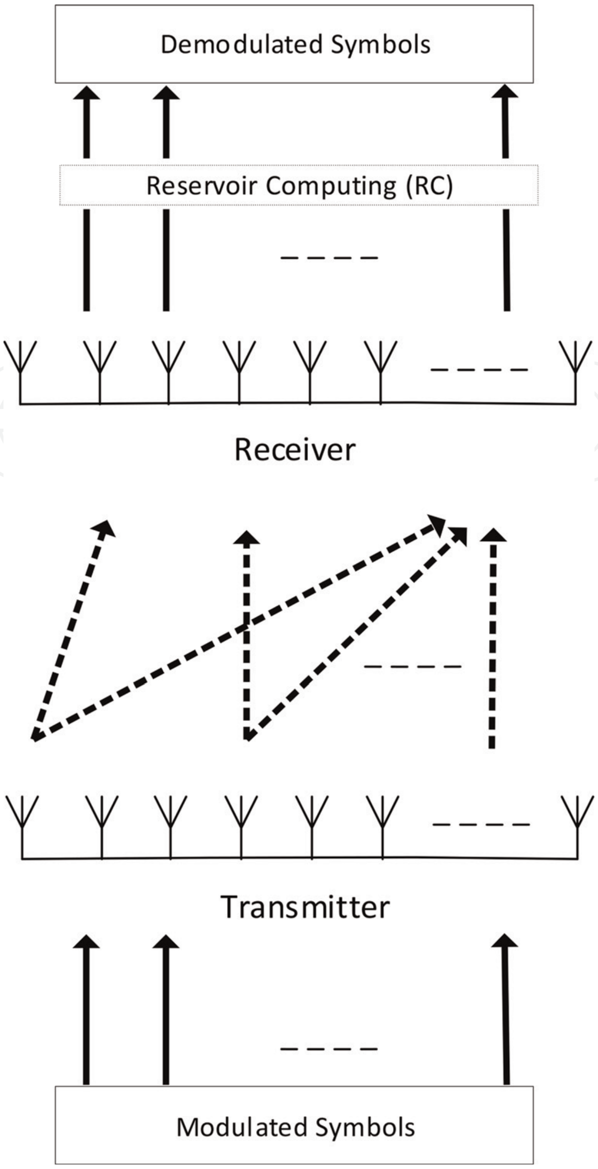

Figure 8.
Symbol detection framework.

where k is the index of interested user; and the remained terms are treated as the interference to the k th user. Given a user index k, the symbol detection is conducted by learn a RC by solving

$$\min_{\mathcal{RC}_k} L\left(\mathcal{RC}_k(\boldsymbol{y}(t)), x_k(t)\right). \tag{19}$$

The symbol detection requires learning k RCs, correspondingly. The trained RCs generate estimated symbols for each stream independently.

Moreover, an input buffer can be incorporated to further improve the symbol detection performance as proposed in [31]. To this end, the input of RC at time t_0 is a batch $\{\boldsymbol{y}(t)\}_{t=t_0}^{t_0+T}$, where T is the length of the buffer.

4.3 One layer learning

We consider the special case when the output is only with one layer. According to the dynamic equation of inner states, denoted as $\{\boldsymbol{s}(t)\}_{t=0}^{T_a-1}$, where T_a is the length of the training data [56], by stacking the states into a matrix $\boldsymbol{S}:=[\boldsymbol{s}(0), \boldsymbol{s}(1), \cdots, \boldsymbol{s}(T_a-1)]$. The output weights can be updated according to

$$\min_{\boldsymbol{W}} \|\boldsymbol{W}\boldsymbol{S} - \overline{\boldsymbol{X}}\|_2 \tag{20}$$

where $\overline{\boldsymbol{X}} \in \mathbb{C}^{N \times T_a}$ is the target waveform at transmitter side, in which N denotes the number of streams; and $\boldsymbol{W}$ is the output layer to be learned. Accordingly, the target waveform $\overline{\boldsymbol{X}}$ can be chosen as the time domain presentation of scattered pilots or comb pilots. For the target of scattered pilots, the (i, t) th entry of $\overline{\boldsymbol{X}}$ is defined as

$$x_i(t) = \sum_{p=0}^{\infty} \sum_{n \in \Omega_p} g\left(t - pT_s\right) s_i[n,p] e^{j\pi\left(nf_0 + f_c\right)t} \tag{21}$$

where Ω_p stands for the index of the sub-carriers selected as pilots in the pth OFDM symbol. Specially, for the comb pilots, Ω_P is defined as all the subcarriers at a certain OFDM symbol or several subarriers across all OFDM symbols.

For solving the problem (20), $\boldsymbol{W}$ can be calculated once whole batch of training data are collected, which is through the following pseudo-inverse operation

$$\hat{\boldsymbol{W}} = \overline{\boldsymbol{X}}\boldsymbol{S}^{+} \tag{22}$$

or thorough an online version, such as gradient descent or recursive least squares [57]. For multiple output layers, it follows the same method as multiple layers feed-forward neural networks via the forward backward propagation procedure [58].

4.4 Simulation results

In **Figure 9**, it demonstrates the BER performance of reservoir computing-based symbol detection methods: simple echo state networks (ESN) and echo state networks with windows (WESN) to the conventional methods: linear minimum mean

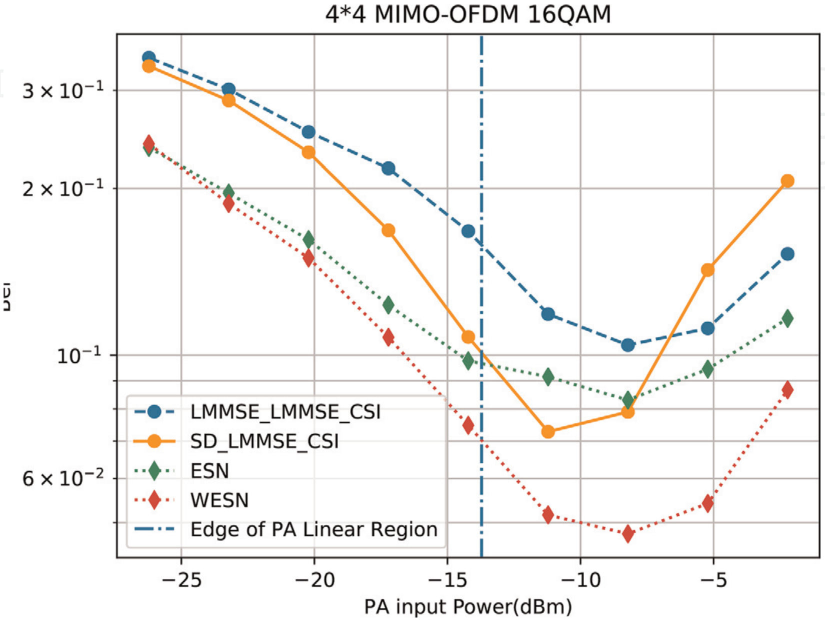

Figure 9.
BER comparison of reservoir computing-based symbol detection methods (ESN and ESN) to conventional methods (LMMSE and sphere decoding).

squared error (LMMSE) and sphere decoding (SD). For the conventional methods, the CSI is obtained by LMMSE channel estimation [59, 60]. Here, we also consider the impact by PA non-linearity at the transmitter side. When the transmitted signal goes throughout the nonlinear region of PA, the signal suffers strong distortion, which can lead to a poor BER performance. Meanwhile, from this figure, we can observe the learning-based methods perform the best at low SNR regime and nonlinear region. This is because conventional methods rely on accurate CSI, which cannot be obtained in these two cases, while learning-based methods are robust against the model-based methods.

5. Conclusion

In this chapter, the emerging applications of spiking DFRs and ESNs were explored. We introduced the combination of spiking neurons, DFRs, and MLPs as the main platform to detect FDI attacks in smart grids. Our simulation results showed that spiking DFR + MLP outperforms SNN, and MLP in terms of ***accuracy*** and ***F1***, respectively. The combination of DFRs and spiking neurons is capable of mapping the data to high dimensional space and capturing the spatio-temporal correlations, which exist between different components of smart grids. The effect of delay value on the performance of DFR was also studied in this chapter. We showed that DFRs can show high dimensional behaviors only for the delay values that make them operate at the edge of chaos. The computational complexity of our introduced model was also studied. In the use case of ESN for MIMO-OFDM symbol detection, we see this learning-based framework can perform better than conventional channel model-based methods when the obtained channel information is imperfect or model mismatch exists. The cost of learning is very few, i.e., it does not require a large size of pilots, which permits the application of this technique in practical system.

Acknowledgements

The work of K. Hamedani, L. Liu and Z. Zhou are supported in part by the U.S. National Science Foundation under grants ECCS-1802710, ECCS-1811497, CNS-1811720, and CCF-1937487.

Author details

Kian Hamedani*, Zhou Zhou, Kangjun Bai and Lingjia Liu
Electrical and Computer Engineering Department, Virginia Tech, Blacksburg, USA

*Address all correspondence to: hkian@vt.edu

References

[1] Li J, Liu L, Zhao C, Hamedani K, Atat R, Yi Y. Enabling sustainable cyber physical security systems through neuromorphic computing. IEEE Transactions on Sustainable Computing. 2017;**3**(2):112-125

[2] Atat R, Liu L, Chen H, Wu J, Li H, Yi Y. Enabling cyber-physical communication in 5g cellular networks: Challenges, spatial spectrum sensing, and cyber-security. IET Cyber-Physical Systems: Theory and Applications. 2017; **2**(1):49-54

[3] Atat R, Liu L, Ashdown J, Medley MJ, Matyjas JD, Yi Y. A physical layer security scheme for mobile health cyber-physical systems. IEEE Internet of Things Journal. 2017;**5**(1):295-309

[4] Li Y, Ng BL, Trayer M, Liu L. Automated residential demand response: Algorithmic implications of pricing models. IEEE Transactions on Smart Grid. 2012;**3**(4):1712-1721

[5] Atat R, Liu L, Wu J, Ashdown J, Yi Y. Green massive traffic offloading for cyber-physical systems over heterogeneous cellular networks. ACM/ Springer Journal of Mobile Networks and Applications. 2018;**24**(4):1-9

[6] Atat R, Liu L, Wu J, Li G, Ye C, Yang Y. Big data meet cyber-physical systems: A panoramic survey. IEEE Access. 2018;**6**:73603-73636

[7] Atat R, Liu L, Yi Y. Privacy protection scheme for ehealth systems: A stochastic geometry approach. In: 2016 IEEE Global Communications Conference (GLOBECOM); IEEE; 2016. pp. 1-6

[8] Wang X, Liu L, Zhu L, Tang T. Joint security and QoS provisioning in train-centric CBTC systems under sybil attacks. IEEE Access. 2019;**7**: 91169-91182

[9] Mosleh S, Sahin C, Liu L, Zheng R, Yi Y. An energy efficient decoding scheme for nonlinear MIMO-OFDM network using reservoir computing. In: 2016 International Joint Conference on Neural Networks (IJCNN); IEEE; 2016. pp. 1166-1173

[10] Zhao C, Danesh W, Wysocki BT, Yi Y. Neuromorphic encoding system design with chaos based CMOS analog neuron. In: 2015 IEEE Symposium on Computational Intelligence for Security and Defense Applications (CISDA); IEEE; 2015. pp. 1-6

[11] Zhao C, Li J, Liu L, Koutha LS, Liu J, Yi Y. Novel spike based reservoir node design with high performance spike delay loop. In: Proceedings of the 3rd ACM International Conference on Nanoscale Computing and Communication; ACM; 2016. p. 14

[12] Bay K, An Q, Yi Y. Deep-DFR: A memristive deep delayed feedback reservoir computing system with hybrid neural network topology. In: Proceedings of the 56th Annual Design Automation Conference 2019; ACM; 2019. p. 54

[13] Esser SK, Merolla PA, Arthur JV, Cassidy AS, Appuswamy R, Andreopoulos A, et al. Convolutional networks for fast, energy-efficient neuromorphic computing. Proceedings of the National Academy of Sciences. 2016;**113**(41):11441-11446

[14] Li J, Zhao C, Hamedani K, Yi Y. Analog hardware implementation of spike-based delayed feedback reservoir computing system. In: 2017 International Joint Conference on Neural Networks (IJCNN); IEEE; 2017. pp. 3439-3446

[15] Li J, Bay K, Liu L, Yi Y. A deep learning based approach for analog hardware implementation of delayed

feedback reservoir computing system. In: 2018 19th International Symposium on Quality Electronic Design (ISQED); IEEE; 2018. pp. 308-313

[16] Bay K, Li J, Hamedani K, Yi Y. Enabling an new era of brain-inspired computing: Energy-efficient spiking neural network with ring topology. In: 2018 55th ACM/ESDA/IEEE Design Automation Conf. (DAC); 2018. pp. 1-6

[17] Zhao C, Hamedani K, Li J, Yi Y. Analog spike-timing-dependent resistive crossbar design for brain inspired computing. IEEE Journal on Emerging and Selected Topics in Circuits and Systems. 2017;**8**(1):38-50

[18] Hu J, Tang H, Tan KC, Li H, Shi L. A spike-timing-based integrated model for pattern recognition. Neural Computation. 2013;**25**(2):450-472

[19] Zhao C, Wysocki BT, Thiem CD, McDonald NR, Li J, Liu L, et al. Energy efficient spiking temporal encoder design for neuromorphic computing systems. IEEE Transactions on Multi-Scale Computing Systems. 2016;**2**(4): 265-276

[20] Hamedani K, Liu L, Atat R, Wu J, Yi Y. Reservoir computing meets smart grids: Attack detection using delayed feedback networks. IEEE Transactions on Industrial Informatics. 2017;**14**(2): 734-743

[21] Yu Q, Tang H, Tan KC, Li H. Precise-spike-driven synaptic plasticity: Learning hetero-association of spatiotemporal spike patterns. PLoS ONE. 2013;**8**(11):e78318

[22] Zhao C, Li J, An H, Yi Y. Energy efficient analog spiking temporal encoder with verification and recovery scheme for neuromorphic computing systems. In: 2017 18th International Symposium on Quality Electronic Design (ISQED); IEEE; 2017. pp. 138-143

[23] C. Zhao, B. T. Wysocki, Y. Liu, C. D. Thiem, N. R. McDonald, and Y. Yi, “Spike-time-dependent encoding for neuromorphic processors,” ACM Journal on Emerging Technologies in Computing Systems, vol. 12, no. 3, pp. 23:1-23:21, Sep. 2015

[24] Zhao C, Yi Y, Li J, Fu X, Liu L. Interspike-interval-based analog spike-time-dependent encoder for neuromorphic processors. IEEE Transactions on Very Large Scale Integration (VLSI) Systems. 2017;**25**(8): 2193-2205

[25] Hegger R, Bünner MJ, Kantz H, Giaquinta A. Identifying and modeling delay feedback systems. Physical Review Letters. 1998;**81**(3):558

[26] She C, Yang C, Liu L. Energy-efficient resource allocation for MIMO-OFDM systems serving random sources with statistical QoS requirement. IEEE Transactions on Communications. 2015; **63**(11):4125-4141

[27] Almosa H, Mosleh S, Perrins E, Liu L. Downlink channel estimation with limited feedback for FDD multi-user massive MIMO with spatial channel correlation. In: 2018 IEEE International Conference on Communications (ICC); IEEE; 2018. pp. 1-6

[28] Mosleh S, Liu L, Ashdown JD, Perrins E, Turck K. Content-based user association and MIMO operation over cached Cloud-RAN networks. arXiv preprint arXiv:1906.11318; 2019

[29] Tse D, Viswanath P. Fundamentals of Wireless Communication. Cambridge University Press; 2005

[30] Shafin R, Liu L, Zhang J, Wu Y-C. DoA estimation and capacity analysis for 3-D millimeter wave massive-MIMO/FD-MIMO OFDM systems. IEEE Transactions on Wireless Communications. 2016;**15**(10): 6963-6978

[31] Zhou Z, Liu L, Chang H-H. Learn to demodulate: MIMO-OFDM symbol detection through downlink pilots. arXiv preprint arXiv:1907.01516; 2019

[32] Atat R, Ma J, Chen H, Lee U, Ashdown J, Liu L. Cognitive relay networks with energy and mutual-information accumulation. In: IEEE INFOCOM 2018-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS); IEEE; 2018. pp. 640-644

[33] Mahmood FE, Perrins ES, Liu L. Energy consumption vs. bit rate analysis toward massive MIMO systems. In: 2018 IEEE International Smart Cities Conference (ISC2); IEEE; 2018. pp. 1-7

[34] Porwal R, Agrawal H, Vyas R. MIMO OFDM space time coding-spatial multiplexing increasing performance and spectral efficiency in wireless systems. International Journal for Scientific Research and Development. 2014;**2**(06):2321-0613

[35] Shafin R, Liu L, Li Y, Wang A, Zhang J. Angle and delay estimation for 3-D massive MIMO/FD-MIMO systems based on parametric channel modeling. IEEE Transactions on Wireless Communications. 2017;**16**(8):5370-5383

[36] Shafin R, Liu L, Zhang J. DoA estimation and RMSE characterization for 3D massive-MIMO/FD-MIMO OFDM system. In: 2015 IEEE Global Communications Conference (GLOBECOM); IEEE; 2015. pp. 1-6

[37] Shafin R, Jiang M, Ma S, Piazzi L, Liu L. Joint parametric channel estimation and performance characterization for 3D massive MIMO OFDM systems. In: 2018 IEEE International Conference on Communications (ICC); IEEE; 2018. pp. 1-6

[38] Shafin R, Liu L. DoA estimation and performance analysis for multi-cell multi-user 3D mmwave massive-MIMO OFDM system. In: 2017 IEEE Wireless Communications and Networking Conference (WCNC); IEEE; 2017. pp. 1-6

[39] Liu L, Chen R, Geirhofer S, Sayana K, Shi Z, Zhou Y. Downlink MIMO in LTE-advanced: SU-MIMO vs. MU-MIMO. IEEE Communications Magazine. 2012;**50**(2):140-147

[40] Mahmood FE, Perrins ES, Liu L. Modeling and analysis of energy consumption for MIMO systems. In: 2017 IEEE Wireless Communications and Networking Conference (WCNC); IEEE; 2017. pp. 1-6

[41] Shafin R, Liu L, Zhang JC. On the Channel Estimation for 3D Massive MIMO Systems. E-LETTER; 2014

[42] Ozdemir MK, Arslan H. Channel estimation for wireless OFDM systems. IEEE Communication Surveys and Tutorials. 2007;**9**(2):18-48

[43] Shafin R, Liu L, Ashdown J, Matyjas J, Zhang J. On the channel estimation of multi-cell massive FD-MIMO systems. In: 2018 IEEE International Conference on Communications (ICC); IEEE; 2018. pp. 1-6

[44] Shafin R, Chen H, Nam YH, Hur S, Park J, Reed J, et al. Self-tuning sectorization: Deep reinforcement learning meets broadcast beam optimization. arXiv preprint arXiv: 1906.06021; 2019

[45] Shafin R, Liu L. Multi-cell multi-user massive FD-MIMO: Downlink precoding and throughput analysis. IEEE Transactions on Wireless Communications. 2018;**18**(1):487-502

[46] Liodakis G, Arvanitis D, Vardiambasis I. Neural network-based digital receiver for radio

communications. WSEAS Transactions on Systems. 2004;**3**(10):3308-3313

[47] Cai H, Zhao X-h. MIMO-OFDM channel estimation based on neural network. In: 2010 6th International Conference on Wireless Communications Networking and Mobile Computing (WiCOM); IEEE; 2010. pp. 1-4

[48] Shafin R, Liu L, Chandrasekhar V, Chen H, Reed J, et al. Artificial intelligence-enabled cellular networks: A critical path to beyond-5g and 6g. arXiv preprint arXiv:1907.07862; 2019

[49] Sarma KK, Mitra A. Modeling MIMO channels using a class of complex recurrent neural network architectures. AEU International Journal of Electronics and Communications. 2012;**66**(4):322-331

[50] Routray G, Kanungo P. Rayleigh fading MIMO channel prediction using RNN with genetic algorithm. In: International Conference on Computational Intelligence and Information Technology; Springer; 2011. pp. 21-29

[51] Chang H-H, Song H, Yi Y, Zhang J, He H, Liu L. Distributive dynamic spectrum access through deep reinforcement learning: A reservoir computing-based approach. IEEE Internet of Things Journal. 2018;**6**(2): 1938-1948

[52] Mahmood F, Perrins E, Liu L. Energy-efficient wireless communications: From energy modeling to performance evaluation. IEEE Transactions on Vehicular Technology. 2019;**68**(8):7643-7654

[53] Kim J, Tong L, Thomas RJ. Dynamic attacks on power systems economic dispatch. In: 2014 48th Asilomar Conference on Signals, Systems and Computers; IEEE; 2014. pp. 345-349

[54] Zimmerman RD, Murillo-Sánchez CE, Thomas RJ, et al. Matpower: Steady-state operations, planning, and analysis tools for power systems research and education. IEEE Transactions on Power Apparatus and Systems. 2011;**26**(1):12-19

[55] Mosleh S, Liu L, Sahin C, Zheng YR, Yi Y. Brain-inspired wireless communications: Where reservoir computing meets MIMO-OFDM. IEEE Transactions on Neural Networks and Learning Systems. 2017;**29**(10): 4694-4708

[56] Shafin R, Liu L, Ashdown J, Matyjas J, Medley M, Wysocki B, et al. Realizing green symbol detection via reservoir computing: An energy-efficiency perspective. In: 2018 IEEE International Conference on Communications (ICC); IEEE; 2018. pp. 1-6

[57] Jaeger H. Adaptive nonlinear system identification with echo state networks. In: Advances in Neural Information Processing Systems; 2003. pp. 609-616

[58] Hecht-Nielsen R. Theory of the backpropagation neural network. In: Neural Networks for Perception. Elsevier; 1992. pp. 65-93

[59] Cheng L, Wu Y-C, Ma S, Zhang J, Liu L. Channel estimation in full-dimensional massive MIMO system using one training symbol. In: 2017 IEEE 18th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC); IEEE; 2017. pp. 1-5

[60] Danesh W, Zhao C, Wysocki BT, Medley MJ, Thawdar NN, Yi Y. Channel estimation in wireless OFDM systems using reservoir computing. In: 2015 IEEE Symposium on Computational Intelligence for Security and Defense Applications (CISDA); IEEE; 2015. pp. 1-5