IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 31, 2017, accepted September 22, 2017, date of publication September 26, 2017,
date of current version October 25, 2017.

Digital Object Identifier 10.1109/ACCESS.2017.2756992

Byzantine Defense in Collaborative Spectrum Sensing via Bayesian Learning

GUANGMING NIE[1], GUORU DING[1,2], (Senior Member, IEEE), LINYUAN ZHANG[1], AND QIHUI WU[3], (Senior Member, IEEE)

[1]College of Communications Engineering, PLA University of Science and Technology, Nanjing 210000, China
[2]National Mobile Communications Research Laboratory, Southeast University, Nanjing 210018, China
[3]College of Electronic and Information Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China

Corresponding authors: Guoru Ding (dr.guoru.ding@ieee.org) and Qihui Wu (wuqihui2014@sina.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61501510 and Grant 61631020, in part by the Natural Science Foundation of Jiangsu Province under Grant BK20150717, in part by the China Postdoctoral Science Foundation Funded Project under Grant 2016M590398, and in part by Jiangsu Planned Projects for Postdoctoral Research Funds under Grant 1501009A.

ABSTRACT Collaborative spectrum sensing (CSS) enables secondary users in cognitive radio networks to collaboratively explore spectrum holes as well as protecting the primary user from being interfered. Unfortunately, the emergence of spectrum sensing data falsification (SSDF) attack, also known as the Byzantine attack, brings significant threat to the reliability of the CSS. Majority of the existing studies on Byzantine defense can be divided into two categories: one is directly to make the judgment based on the current spectrum sensing data, while the other uses the historical spectrum sensing data to update sensors' reputation. The first category of studies does not take the historical spectrum sensing data into account, while most of the second category of studies are heuristic in nature. In this paper, we invoke Bayesian learning to design Byzantine defense schemes. First, we develop a Bayesian offline learning algorithm by considering one practical challenge that the ground-truth spectrum state is unavailable for training. Then, we develop a Bayesian online learning algorithm by considering the case that the sensors' attribute may be time-varying. In addition, we present simulations to show the performance of the proposed defence algorithms.

INDEX TERMS Cognitive radio networks, collaborative spectrum sensing, Byzantine attack, Bayesian online learning.

I. INTRODUCTION

Increasing demand for wireless communication in various areas of human life has brought an exponential increase in the number of wireless services. Such increase has resulted in spectrum scarcity since the electromagnetic spectrum has become too crowded to incorporate the upcoming wireless services [1]. In order to make full use of spectrum resources, cognitive radio technology has been proposed [2], where the secondary user (SU) first performs the spectrum sensing process, and then accesses the spectrum based on the spectrum sensing result. To overcome the negative impacts of noise, path loss, shadowing, and fading, cooperative spectrum sensing (CSS) among multiple spatially-dispersed SUs has received great research attention [3].

However, in order to maximize their own interests, some SUs participating the CSS process may report false information to mislead the decision-making. Usually, the case of malicious falsification in the literature is well known as the spectrum sensing data falsification attack, or Byzantine attack [4]–[6]. To eliminate the impact of Byzantine attack, increasing studies on Byzantine defense have been reported (see, e.g., [7]–[17]). Generally, the existing Byzantine defense in the literature can be grouped into two classes: current data based defense (CDbD) (see, e.g., [5], [18], [19]) and reputation updating based defense (RUbD) (see, e.g., [1], [20]). Based on the current spectrum sensing data, the CDbD makes a judgment on the sensors' attributes without using the historical data. In practice, since the sensors' behavior is relevant, the historical data has a certain value on the judgment. On the other hand, in the RUbD, the historical spectrum sensing data is used to update sensors' reputation. However, most of the studies are heuristic in nature.

Motivated by the fact that the existing work has strong limitations, for example, the historical spectrum sensing data is not used in Byzantine defense or the historical spectrum

TABLE 1. Notations and symbols used in this paper.

Symbols	Definitions
$\mathcal{H}$	The channel's real state
N	The number of sensors
p_f	The sensor's false-alarm probability in sensing process
p_d	The sensor's detection probability in sensing process
p_c	The sensor's sensing error probability
p'_f	The sensor's false-alarm probability after sensing process
p'_d	The sensor's detection probability after sensing process
p_a	The probability of a Byzantine attacker conducting attacks when the channel's state is locally decided as idle
p_b	The probability of a Byzantine attacker conducting attacks when the channel's state is locally decided as busy
p_f^b	The false-alarm probability of a Byzantine attacker after Byzantine process
p_d^b	The detection probability of a Byzantine attacker after Byzantine process
$\mathcal{V}$	The sensing results
p_e	The transmit error probability in the reporting process
$\mathcal{U}$	The sensors' reports
P_f^H	The false-alarm probability of an honest sensor
P_d^H	The detection probability of an honest sensor
P_f^B	The false-alarm probability of a Byzantine sensor
P_d^B	The detection probability of a Byzantine sensor
$\mathcal{F}$	The global decisions
w_i	The i^{th} sensor's attribute
k_1^i	The i^{th} sensor's weight after Bayesian offline learning
k_2^i	The i^{th} sensor's weight after current data learning
k_i	The i^{th} sensor's weight after Bayesian online learning
D_T	The historical sample data

sensing data is used heuristically, this paper firstly focuses on the issue of applying the Bayesian learning to Byzantine defense. Bayesian learning is well known as a powerful tool to learn from the given historical spectrum sensing data [21]. However, there are several limitations in direct application of Bayesian learning: i) When the Byzantine attacker in the given historical spectrum sensing data did not exhibit its attack characteristics, it would probably not be able to recognize the attacker, which has a terrible impact on the decision making. Similarly, because of the terrible sensing environment, the honest sensors might show relatively poor sensing performance, leading to the isolation of the honest sensors. As a result, the performance of CSS would be worse. ii) Since the the results of the Bayesian offline learning are fixed after the trading process, it could not be changed once the sensor's attribute is determined. In fact, the sensor's performance may be changing with time. Sometimes, Byzantine attackers will show good sensing performance, and the honest sensors will show poor sensing performance. While the sensor has no opportunity to re-enter the spectrum sensing system once it is identified as attacker in majority of the existing studies. Bayesian online learning is a possible solution to this problem. It not only takes the historical data into consideration, but also uses the current data [22], [23]. By using the current data to dynamically adjust the results obtained by the historical data, which is a good solution to the time-varying sensor attribute attack.

The main idea of applying Bayesian learning to Byzantine defense in this paper is that it can solve the historical spectrum sensing data utilization and the time-varying sensor attribute attack. The contributions of this paper are:

- We develop a Bayesian offline learning algorithm by considering one practical challenge that the ground-truth spectrum state is unavailable for training.
- We present a Bayesian online learning framework to dynamically identify Byzantine attackers. The framework is divided into two parts, the historical data learning part and the current data learning part. The vector of sensors' weight is updated by considering both the historical spectrum sensing data and the current spectrum sensing data into consideration. What's more, we propose a Byzantine attack behavior recognition algorithm based on the proposed framework. The proposed algorithm is more accurate and sensitive to the identification of Byzantine attacks than offline learning.
- We present simulations to show the effectiveness of the proposed Bayesian learning-based Byzantine defense framework and algorithms.

The rest of this paper is organized as follows. Section II presents the process of collaborative spectrum sensing.

Section III introduces the Byzantine defense via Bayesian offline learning. Then, we introduce the Byzantine defense with time-varying sensor attribute via Bayesian online learning in Section IV. Furthermore, we give the conclusion in Sections V.

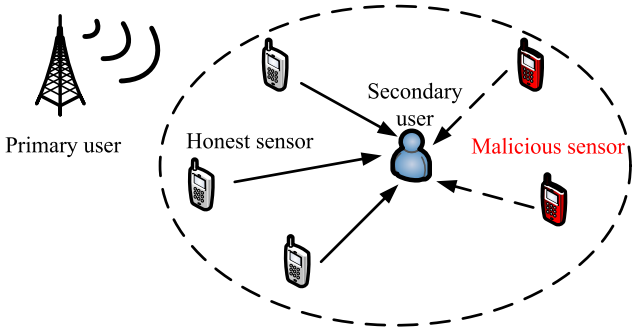


FIGURE 1. System model.

II. THE PROCESS OF COLLABORATIVE SPECTRUM SENSING

In this section, we present a CSS model under Byzantine attack. As depicted in Fig. 1, a CSS system consists of N spectrum sensors sensing the occupancy state of a licensed spectrum band in order to opportunistically access the licensed spectrum whenever it is idle. We consider the case that a SU wants to exploit spectrum holes but its sensing performance is undesirable due to serious shadowing. As a result, it is a feasible solution to ask for help from other neighboring spectrum sensors. Based on its observation, each sensor solves a hypothesis testing problem, in which a decision is made between the hypothesis $\mathcal{H}_1$ (The channel's state is busy) and hypothesis or $\mathcal{H}_0$ (The channel's state is idle). The one-bit decision of sensor i is denoted by v_i, $i = 1, 2, ..., N$. Then, sensor i sends its one-bit output v_i to the SU who also plays as a fusion center (FC) role.

A. SENSING UNCERTAINTY

To decide whether the primary user is present or not, each spectrum sensor makes energy detection and decides between $\mathcal{H}_0$ and $\mathcal{H}_1$. For each sensor, the spectrum sensing is generally formulated as a binary hypotheses testing problem as follows [24]:

$$\begin{cases} \mathcal{H}_0 : r(t) = n(t), \\ \mathcal{H}_1 : r(t) = \alpha \cdot s(t) + n(t), \end{cases} \tag{1}$$

where $r(t)$ is the received signal at that sensor in time t, $s(t)$ is the PU's transmit signal, α is the channel gain, and $n(t)$ denotes the additive white Gaussian noise.

With an energy detector, the collected energy observation can be given as $x_E = \sum_{t=1}^{2U} |r(t)|^2$, where $U = T \cdot W$ is the time-bandwidth product. According to the central limit theorem, when U is sufficiently large (e.g., $U \gg 10$), x_E can be well approximated as a Gaussian random variable under both hypotheses $\mathcal{H}_0$ and $\mathcal{H}_1$ as follows [25]:

$$\begin{cases} \mathcal{H}_0 : x_E \sim N(\mu_0, \sigma_0^2), \\ \mathcal{H}_1 : x_E \sim N(\mu_1, \sigma_1^2), \end{cases} \tag{2}$$

where $\mu_0 = 2U$, $\sigma_0^2 = 4U$, $\mu_1 = 2U(\beta + 1)$, $\sigma_1^2 = 4U(2\beta + 1)$ and β is the received signal to noise ratio (SNR) of the sensor.

Each sensor uses an energy detection scheme with an identical local threshold λ for binary hypothesis testing between $\mathcal{H}_0$ and $\mathcal{H}_1$ [5]. As a result, the binary decision v_i of the SU can be obtained as follows:

$$x_E \underset{v_i=1}{\overset{v_i=0}{\gtrless}} \lambda. \tag{3}$$

Here, we introduce two metrics, namely, the probability of false alarm p_f and the probability of detection p_d to describe the performance of spectrum sensing, and the corresponding definitions are as follows:

$$\begin{cases} p_f \triangleq P(v_i = 1|\mathcal{H}_0) = Q\left(\dfrac{\lambda - \mu_0}{\sigma_0}\right), \\ p_d \triangleq P(v_i = 1|\mathcal{H}_1) = Q\left(\dfrac{\lambda - \mu_1}{\sigma_1}\right), \end{cases} \tag{4}$$

where $Q(z) = (1/\sqrt{2\pi}\ ) \int_z^\infty \exp(-(x^2/2\ ))dx$ is the complement distribution function for the normal distribution with zero mean and unit variance. Further, the missed detection probability can be given as $p_m = 1 - p_d$.

What we have to point out is that we introduce the sensing error probability p_c in order the existence of realistic situation, imperfect sensing. Consequently, the corresponding detection probability and false alarm probability can be formulated as follows, respectively,

$$\begin{cases} p_d' = p_d \cdot (1 - p_c) + (1 - p_d) \cdot p_c, \\ p_f' = p_f \cdot (1 - p_c) + (1 - p_f) \cdot p_c. \end{cases} \tag{5}$$

B. BYZANTINE ATTACK

As aforementioned, after performing local hypothesis testing, each spectrum sensor transmits its one-bit hard decision u_i to the fusion center. An honest sensor will report its original local decision v_i to the fusion center, i.e., $u_i = v_i$, for data fusion. However, for a Byzantine attacker, u_i may not be the same as v_i. That is to say, the Byzantine attacker will falsify its local decision with a certain probability.

Specifically, for the Byzantine attack, we introduce two probabilities, i.e., the probability of changing the decision of the primary user being absent to that of the primary user being present p_a, $p_a = \Pr(u_i = 0|v_i = 1)$ and the probability of changing the decision of the primary user being absent to that of the primary user being present p_b, $p_b = \Pr(u_i = 1|v_i = 0)$. The attack probabilities p_a and p_b can range from 0 to 1, with 0 and 1 denoting two extreme cases, never-attack and always-attack, respectively.

Consequently, the corresponding sensing performance of a Byzantine attacker is formulated as follows

$$\begin{cases} p_d^b = p_d' \cdot (1 - p_a) + (1 - p_d') \cdot p_b, \\ p_f^b = p_f' \cdot (1 - p_a) + (1 - p_f') \cdot p_b. \end{cases} \tag{6}$$

C. IMPERFECT REPORTING

In practice, owing to the hostile transmission environment, the reporting channel between a sensor and the fusion center is imperfect. To characterize this effect, we introduce a transmission error probability p_e for the data reporting process. Now, we can derive the corresponding probabilities of false alarm P_f^H, P_f^B and the probabilities of detection P_d^H, P_d^B for both the honest sensors and Byzantine sensors, respectively.

For honest sensors:

$$\begin{cases} P_f^H = p_f' \cdot (1 - p_e) + (1 - p_f') \cdot p_e, \\ P_d^H = p_d' \cdot (1 - p_e) + (1 - p_d') \cdot p_e. \end{cases} \tag{7}$$

For Byzantine sensors:

$$\begin{cases} P_f^B = p_f^b \cdot (1 - p_e) + (1 - p_f^b) \cdot p_e, \\ P_d^B = p_d^b \cdot (1 - p_e) + (1 - p_d^b) \cdot p_e. \end{cases} \tag{8}$$

D. DATA FUSION

Based on the received local decisions of each spectrum sensor, the fusion center makes a global decision $\mathcal{F}$ about the state of the PU ($\mathcal{H}_0$ or $\mathcal{H}_1$). The main fusion rules include L out of N rule [26] and likelihood ratio test (LRT)-based rule [27], among which the LRT-based rule is the optimal data fusion rule which jointly utilizes the sensing reports and the average sensing performance of each sensor. Relatively, L out of N rule is simple to implement and suitable for the case that all sensors have the same sensing performance.

III. BYZANTINE DEFENSE VIA BAYESIAN OFFLINE LEARNING

In this section, we first give the theoretical derivation of Bayesian offline learning. Then, we give the simulation results of Bayesian offline learning-based defense algorithm.

A. BAYESIAN OFFLINE LEARNING

1) PRIOR DISTRIBUTION

In the CRN, sensor i corresponds to an attribute index w_i. Specifically, the honest user's attribute is set to 1, and the malicious sensor's attribute is set to -1. Besides, each sensor has a corresponding weight k_i to balance its trustworthiness. The weight value varies between 0 and 1. Here, we assume that the initial value of k_i is 1/2. The statistical independence of the individual examples results in a multiplicative form for the likelihood of the sensor's attribute:

$$p(\mathbf{w}) = \prod_i [k_i \delta(w_i - 1) + (1 - k_i)\,\delta(w_i + 1)]. \tag{9}$$

2) SAMPLE DATA

In this paper, the sample data is denoted by $D_T = \left\{\left(\mathcal{F}^t, \mathbf{u}^t\right), 1 \le t \le T\right\}$, where $\mathbf{u}^t = \{u_i, 1 \le i \le N\}$ is the reported by all users to the data fusion center in the t time slot, and $\mathcal{F}^t \in \{0, 1\}$ is the corresponding outcome of the primary user's channel state made by data fusion during the t time slot. Due to the examples are independently drawn from a distribution $p((\mathcal{F}, \mathbf{u})|\mathbf{w})$. The statistical independence of the individual examples results in a multiplicative form for the likelihood of the training set:

$$p(D_T|\mathbf{w}) = \prod_t p\left(\left(\mathcal{F}^t, \mathbf{u}^t\right)|\mathbf{w}\right). \tag{10}$$

We write $p((\mathcal{F}, \mathbf{u})|\mathbf{w}) = p(\mathcal{F}|\mathbf{w}, \mathbf{u})\,p(\mathbf{u})$, where $p(\mathcal{F}|\mathbf{w}, \mathbf{u})$ models the input-output relation, while the input distribution $p(\mathbf{u})$ is independent of $\mathbf{w}$.

3) DATA FUSION RULE

In this paper, the data fusion rule $\mathcal{F} \in \{0, 1\}$ can be expressed as follows:

$$\mathcal{F} = f(\mathbf{k}, \mathbf{u}) = \mathrm{U}(\mathbf{k} \cdot \mathbf{u} - \mathrm{N}/4), \tag{11}$$

where $\mathrm{U}(x) = \begin{cases} 1, & x > 0 \\ 0, & x \le 0 \end{cases}$, both $\mathbf{k}$ and $\mathbf{u}$ are N-dimensional vectors with components $\{k_i, 1 \le i \le N\}$ and $\{\mathrm{u}_i, 1 \le i \le N\}$, respectively, N is the total number of sensors participate in CSS, and the error probability of data fusion is introduced by probability g. The likelihood of output $\mathcal{F}$ is thus given by

$$p(\mathcal{F}|\mathbf{w}, \mathbf{u}) = g + (1 - g)\ \delta(\mathcal{F} - f(\mathbf{w}, \mathbf{u})). \tag{12}$$

What we have to point out is that the $\mathcal{F}$ expressed in (12) is only the possible outcome of the channel state, rather than the exact result of the data fusion center.

4) BAYESIAN POSTERIOR PROBABILITY

Bayes rule provides a prescription for writing the posterior distribution $p(\mathbf{w}|D_T)$ in terms of the prior and the likelihood of the sample:

$$p(\mathbf{w}|D_T) = \frac{p(\mathbf{w}) \prod_t p\left(\mathcal{F}^t|\mathbf{w}, \mathbf{u}^t\right)}{\int \prod_t p(\mathcal{F}^t|\mathbf{w}, \mathbf{u}^t) p(\mathbf{w})\ d\mathbf{w}}. \tag{13}$$

The posterior distribution quantifies our knowledge about $\mathbf{w}$ after the observation of the training data D_T, and it is used to compute the predictive probability for each possible output $\mathcal{F} \in \{0, 1\}$ given a new input $\mathbf{u}^{T+1}$:

$$p\left(\mathcal{F}|\mathbf{u}^{T+1} D_T\right) = \int p\left(\mathcal{F}|\mathbf{w}, \mathbf{u}^{T+1}\right)\ p(\mathbf{w}|D_T)\ d\mathbf{w}. \tag{14}$$

Predictions based on the Bayes algorithm are guaranteed to minimize the average prediction error through the choice of output $\mathcal{F}$ which maximizes the above prediction probability for a given input $\mathbf{u}^{T+1}$. For the spectrum sensing problem considered here, the Bayes prediction is given by

$$\mathcal{F}^{\mathrm{Bayes}}\left(\mathbf{u}^{T+1}, D_T\right) = \mathrm{U}\left(\int p(\mathrm{w}|D_T)\ \mathrm{U}\left(\mathbf{w} \cdot \mathbf{u}^{T+1}\right)\ d\mathbf{w}\right). \tag{15}$$

Algorithm 1 Bayesian Offline Learning-based Byzantine Defense Algorithm

Initialize: Set N as the number of sensors, and T as the training time slots. Set $p(\mathbf{w})$ as the prior distribution of the sensors' attributes, and $\mathbf{k}$ as the vector of sensors' weight with the initial value 1/2.
for $t = 1, 2, ..., T$ **do**
  for $i = 1, ..., N$ **do**
    Each sensor performs local spectrum sensing and reports its result u_i to the FC.
  end for
  The FC performs data fusion: $\mathrm{F} = f(\mathbf{k}, \mathbf{u}) = \mathrm{U}(\mathbf{k} \cdot \mathbf{u} - \mathrm{N}/4)$
  for $i = 1, ..., N$ **do**
    IF $u_i \neq \mathcal{F}$, then the *ith* sensor is malicious.
    ELSE IF $u_i = \mathcal{F}$, then the *ith* sensor is honest.
  end for
  Update the distribution of the sensors' attributes $p(\mathbf{w})$ and the vector of sensors's weight $\mathbf{k}$.
end for

Furthermore, the proposed Bayesian offline learning Byzantine attacker identification algorithm is summarized as Algorithm 1.

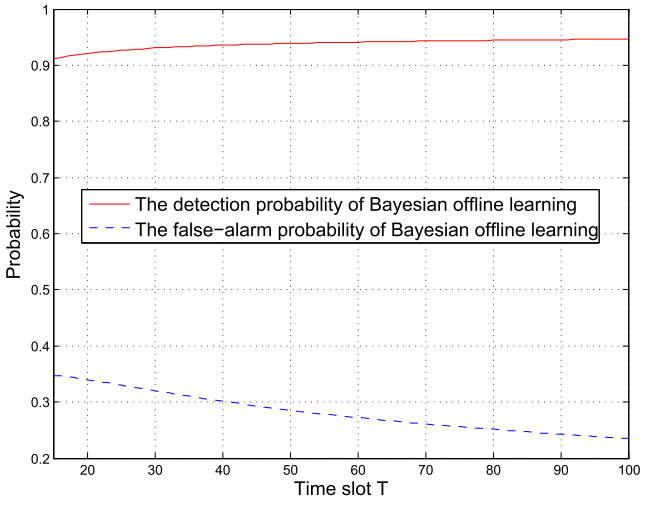


FIGURE 2. The Bayesian offline learning's detection probability and false-alarm probability change with time slots T.

B. SIMULATION RESULTS OF BAYESIAN OFFLINE LEARNING

As shown in Fig. 2, we present the simulation result of the Bayesian offline learning's detection probability and false-alarm probability with time slot T. Obviously, the detection probability increases with the increase of the number of the time slot T, while the false alarm probability decreases with the increase of the number of time slot T. What we can conclude is that the performance of the defense system has been improved through the training of the sample data.

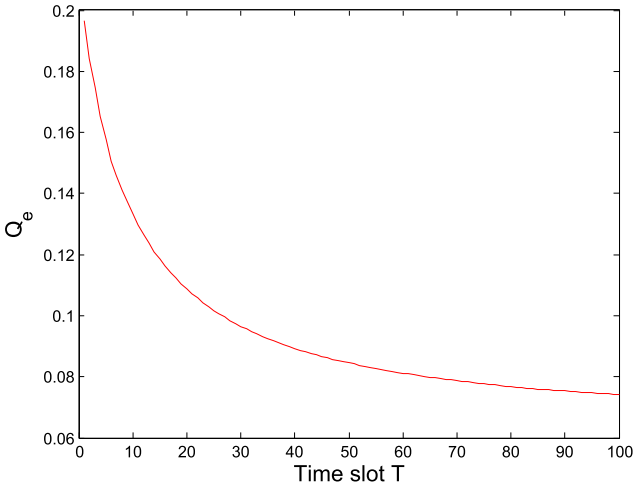


FIGURE 3. The Bayesian offline learning's error probability changes with time slots T.

As shown in Fig. 3, the Bayesian offline learning's error probability Q_e changes with time slot T. And the definition of the error probability is $Q_e \triangleq P(\mathcal{H}_0)P(\mathcal{F} = 1|\mathcal{H}_0) + P(\mathcal{H}_1)P(\mathcal{F} = 0|\mathcal{H}_1)$.Obviously, as the increasing of the time slot T, the error probability Q_e of the defense system is greatly reduced. Namely, the performance of the system is greatly improved.

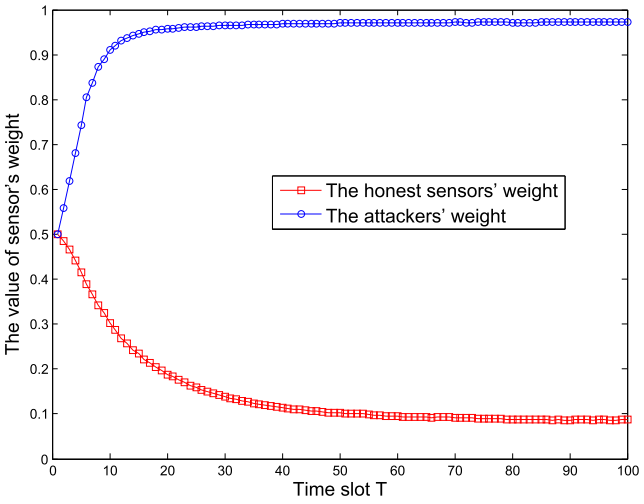


FIGURE 4. The value of sensor's weight changes with time slots T.

Fig. 4 shows the relation of the sensor's weight with the number of the time slots T. As the time slot T increases, the value of the sensor's weight tends to be stable. For the honest sensors, with the cumulative number of samples of learning, the weight gradually increase to 1. On the contrary, for malicious sensors, with the cumulative number of samples of the learning, the network parameters gradually reduced. That is to say, the honest sensors play a more important role in data fusion gradually, while the impact of malicious sensors on the system is getting lower and lower.

IV. BYZANTINE DEFENSE WITH TIME-VARYING SENSOR ATTRIBUTE VIA BAYESIAN ONLINE LEARNING

In this section, we first give the time-varying sensor attribute model. Then, we show the difference between offline learning and online learning. Besides, the dynamic defense framework is introduced. Moreover, we give the Bayesian online learning defense process. And the performance evaluation is shown at last.

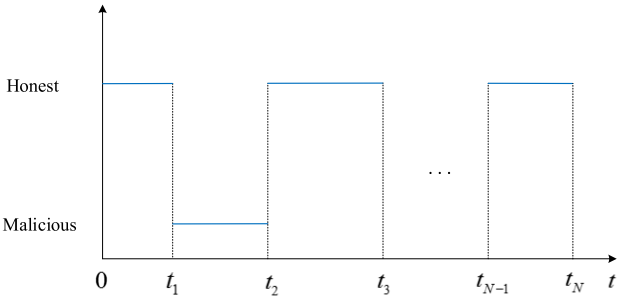


FIGURE 5. Illustration of a Byzantine sensor with time-varying attributes.

A. TIME-VARYING SENSOR ATTRIBUTE MODEL

In this paper, a more common and practical way of attack modeling is considered. The sensors' attributes are time-varying, rather than immutable. As shown in the Fig. 5, sensor's attribute changes between honest sensor and attacker at different times. The reasons for this are as follows: (i) the sensor reports the malicious data for its own interests; (ii) the poor spectrum sensing environment declines sensor's sensing accuracy. These would lead to errors in the results of the CSS process.

Taking this attack mode into account, the Bayesian offline learning approach is not applicable to be used to solve this problem. The reason why the Bayesian offline learning is not suitable is that the Bayesian offline learning only uses the historical data when the decision is made. That is to say, one sensor shows the honest attribute (attack attribute) in the historical data, but it may show the attack attribute (honest attribute) in the working time. As a result, we have to find a new approach to solve this problem. The Bayesian online learning method can be an appropriate solution to this problem. Next, we will introduce the difference between offline learning and online learning.

B. DIFFERENCE BETWEEN OFFLINE LEARNING AND ONLINE LEARNING

As shown in Fig. 6, we give the working diagram of offline learning. Specifically, the horizontal axis represents the time slots and the vertical axis represents the sensors. During each time slot, the larger rectangle represents the attributes of each sensor, where green indicates that the sensor has an honest attribute, and red indicates that the sensor appears as a malicious attribute. Besides, the smaller rectangle is used to represent the value of sensors' weight. The depth of the color represents the difference of the weight. The deeper of the color, the greater of sensors' weight. In offline learning, there is a training process, based on historical data, to attain the value of each sensor's weight. It can be seen from the figure, after t_N time slots of the historical data learning, each sensor's weight is attained. Since the different performance of each sensor in historical data, the final value of each sensor's weight wolud be different. What we need to pay attention to is that the value of each sensor's weight is a fixed value after the training process. It is this feature that makes the offline learning show some drawbacks in practice.

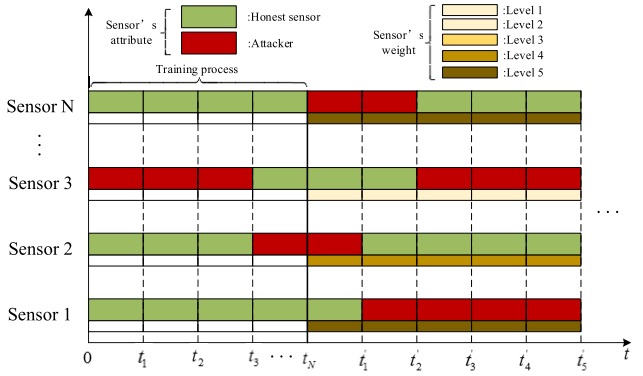


FIGURE 6. Bayesian offline learning.

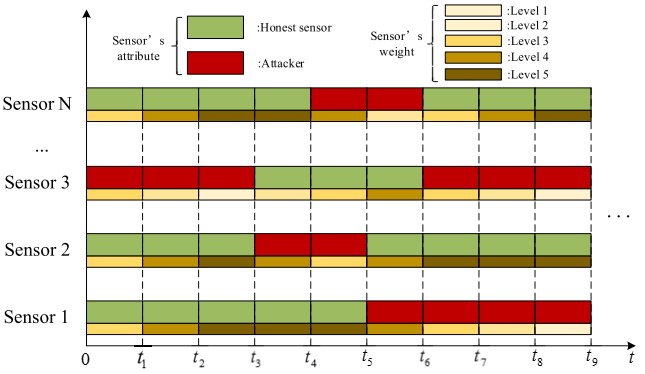


FIGURE 7. Bayesian online learning.

As shown in Fig. 7, we give the working diagram of online learning. As with the offline learning, the horizontal axis represents the time slots and the vertical axis represents the sensors. During each time slot, the larger rectangle represents the attributes of each sensor, where green indicates that the sensor has an honest attribute, and red indicates that the sensor appears as a malicious attribute. Besides, the smaller rectangle is used to represent the value of sensors' weight. The depth of the color represents the difference of the weight. And the deeper of the color, the greater of sensors' weight is. Unlike the offline learning, the value of each sensor's weight in the online learning is no longer fixed. That is to say, as the sensors' attributes change, the value of each sensors' weights change along with it. Specifically, if the sensor exhibits the attack characteristic, its weight is reduced. And if the sensor exhibits the honest characteristic, its weight increases.

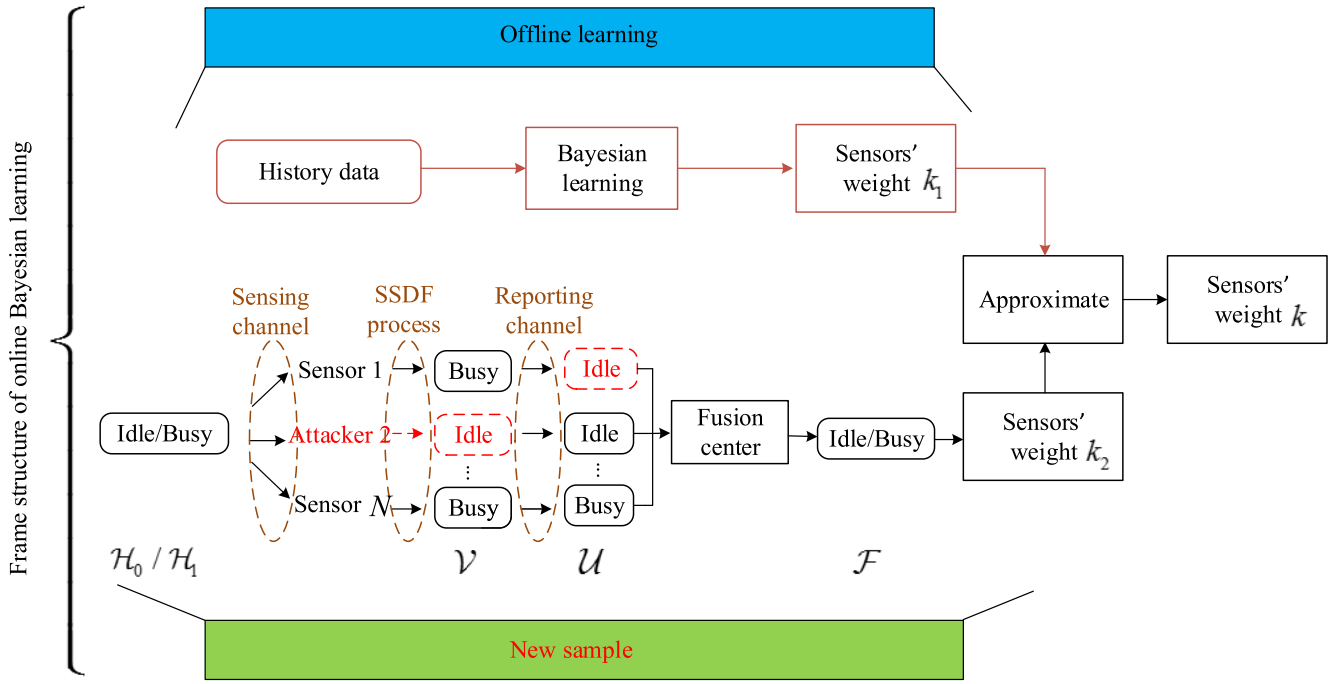


FIGURE 8. The framework of Bayesian learning, where $\mathcal{H}_0/\mathcal{H}_1$ denotes the ground-truth channel state, $\mathcal{V}$ denotes the sensing results, $\mathcal{U}$ denotes the reports of sensors, and $\mathcal{F}$ denotes the results after data fusion.

C. A DYNAMIC DEFENSE FRAMEWORK

In the section III, we give the attack model. Through the analysis of the attack model, we know that the sensors' attributes could change at any time. As a result, the assumption that the sensors' attributes are invariable are no longer reasonable. Therefore, for the new problem, the online learning method shows outstanding advantages. The key idea of Bayesian online learning is that the sensors' weights are dynamically adjusted, combined with historical data and current data.

As shown in Fig. 8, we present the framework of online learning. In general, the proposed framework is divided into two phases, historical data learning phase and new sample learning phase. In the historical data learning phase, we use the Bayesian offline learning method to train the existing historical data. Through the training, we finally attain the value sensors' weight $\mathbf{k}_1$. In new sample learning phase, we get the value of sensor's weight $\mathbf{k}_2$ through the analysis of the current sensing data. The main idea of online learning is that both the results obtained from the historical data and the results obtained from the current data are used to obtain the new weight of each sensor. The advantage of this approach is that the result fits the reality situation by dynamically adjust the value of sensors' weights.

D. BAYESIAN ONLINE LEARNING

Bayesian online learning provides a framework for formulating the Byzantine Identification problem of learning from examples in purely probabilistic terms. In this paper, based on the Bayesian learning, we propose an Byzantine attacker defense framework. Specifically, the sample data is used to train the sensors' attribute and its weights as an offline learning phase. During the online phase, the current data is used to update the sensors' attribute and its weights gained on offline learning phase. By dynamically adjusting the sensors' attribute and its weights, we can not only make full use of the history sample data and the current data, but also eliminate the terrible effects of dynamic change of sensors' attribute. As a result, the whole system's performance is greatly improved. In this section, we firstly introduce the Bayesian offline learning that we employed in Byzantine identification work, and then the Bayesian online learning method is given.

The new problem is how to adapt the Bayesian offline learning summarized in the preceding subsection to obtain an online version. Learning methods based on incorporating all information provided by the data into the current values of the sensors' attribute and its weights are easily adapted onto online versions: it suffices to use the information provided by a new example to update the current values of the sensors' attribute and its weights. In a Bayesian formulation, the information provided by the data is incorporated into the sensors' attribute and its weights, and the sensors' attribute and its weights need to be updated in an online manner when a new example becomes available.

1) ADD THE NEW EXAMPLE

When the new $\left(\mathrm{F}^{T+1}, \mathbf{u}^{T+1}\right)$ is available, update the posterior probability $p\left(\mathbf{w}|D_T\right)$ we derived in preceding subsection combined with the new sample. Mathematically, the new

posterior probability can be written as follow:

$$p\left(\mathbf{w}|D_T, \left(\mathrm{F}^{T+1}, \mathbf{u}^{T+1}\right)\right) = \frac{p\left(\mathrm{F}^{T+1}|\mathbf{w}, \mathbf{u}^{T+1}\right) p\left(\mathbf{w}|D_T\right)}{\int p\left(\mathrm{F}^{T+1}|\mathbf{w}, \mathbf{u}^{T+1}\right) p\left(\mathbf{w}|D_T\right) \, d\mathbf{w}}. \quad (16)$$

2) APPROXIMATE

since we have derived the updated posterior probability $p\left(\mathbf{w}|D_T, \left(\mathrm{F}^{T+1}, \mathbf{u}^{T+1}\right)\right)$, what we need to do next is to parameterize the posterior probability. The parametrization process can be expressed as follow:

$$p\left(\mathbf{w}|D_T, \left(\mathrm{F}^{T+1}, \mathbf{u}^{T+1}\right)\right) \rightarrow p\left(\mathbf{w}|D_{T+1}\right). \quad (17)$$

In the parametrization process, we have a compromise between the historical sample data and the new data. As a result, some of the information provided by the historical sample data and the new data is discarded.

Furthermore, the proposed Bayesian online learning Byzantine attacker identification algorithm is summarized as Algorithm 2.

Algorithm 2 Bayesian Online Learning-based Byzantine Defense Algorithm

1: **Initialize:** Set N as the number of sensors, and T as the working time slots. Set $p(\mathbf{w})$ as the prior distribution of the sensors' attributes, and $\mathbf{k}_1$ as the vector of sensors' weights after learning the historical data, $\mathbf{k}_2$ as the vector of sensors' weights after learning the current data, $\mathbf{k}$ as the vector of sensors' weights.
2: **for** $t = 1, 2, ..., T$ **do**
3: **for** $i = 1, ..., N$ **do**
4: Each sensor performs local spectrum sensing and reports its result u_i to the FC.
5: **end for**
6: The FC performs data fusion: $\mathrm{F} = f(\mathbf{k}, \mathbf{u}) = \mathrm{U}(\mathbf{k} \cdot \mathbf{u} - \mathrm{N}/4)$
7: **for** $i = 1, ..., N$ **do**
8: IF $u_i \neq \mathcal{F}$, then the *ith* sensor is malicious.
9: ELSE IF $u_i = \mathcal{F}$, then the *ith* sensor is honest.
10: **end for**
11: Update the distribution of sensor's attributes $p(\mathbf{w})$ and the vector of sensors's weight $\mathbf{k}_2$.
12: Drive the new vector of sensors's weight $\mathbf{k}$ based on $\mathbf{k}_1$ and $\mathbf{k}_2$.
13: **end for**

V. PERFORMANCE EVALUATION

A. BASIC SIMULATION SETUP

In the simulation, 100 nodes participate in the sensing process. Without loss of generality, the probability of the licensed band being busy is 0.4 and honest sensors' local sensing performance is set as $P_d^H = 0.8$, $P_f^H = 0.2$. For simplicity, the attack probability of Byzantine attackers $p_a = p_b = 0.8$.

To show the effectiveness of the Bayesian online learning algorithm, we perform comparison among the following two schemes:

- Bayesian offline learning algorithm.
- Bayesian online learning algorithm.

B. SIMULATION RESULTS OF BAYESIAN ONLINE LEARNING

In this subsection, we present the simulation results of the Bayesian online learning.

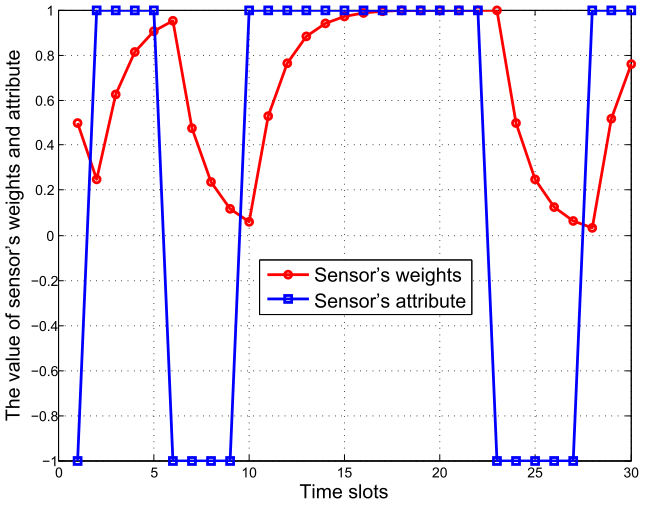


FIGURE 9. The sensor's attribute and corresponding weight change with the time slots T. Without loss of the general, the number of time slots for each attribute is randomly generated.

As shown in Fig. 9, we show the sensor's attribute and corresponding weight change with the time slots. In the given figure, with the sensor's attribute changes, the value of sensor's weight also changes. Specifically, the sensor's attribute changes between the honest and the attack (i.e., w change between 1 and -1), the weight of the sensor's would increase or decrease. When the sensor shows the honest attribute in a certain period of time slots, the value of sensor's weight increase. On the contrary, when the sensor shows the attack attribute in a certain period of time slots, the value of sensor's weight would reduce. In addition, when the sensor lasted longer in a certain attribute, the proposed algorithm show its better performance. For example, in the period of the 10-22-th time slots, the sensor shows the honest attribute. With the increase of the time slots, the value of sensor's weight increase. And the weight reaches the maximum at the 17-th time slot, after which the weight maintains the value in subsequent time slots.

As shown in Fig. 10, we present the Bayesian online learning's and Bayesian offline learning's detection probabilities and false alarm probabilities with sensing time slots. In general, with the increase of sensing time slots, the performance Bayesian online learning gradually improved. On the contrary, the performance Bayesian offline learning gradually become worse. Specifically, on one hand, the false alarm

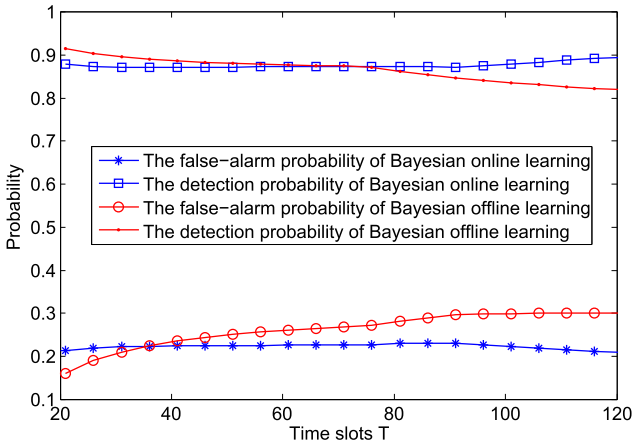


FIGURE 10. **The Bayesian online learning's and Bayesian offline learning's detection probability and false-alarm probability change with time slots *T*.**

probability of offline learning increases with the increase of sensing time slots, while the false alarm probability of online learning decreases with the increase of sensing time slots. On the other hand, the detection probability of offline learning decreases with the increase of sensing time slots, and the detection probability of online learning increases with the increase of sensing time slots. Bayesian online learning has a stronger ability to adapt to changing sensor's attribute.

VI. CONCLUSION

This paper studied the issue of collaborative spectrum sensing against Byzantine attack via Bayesian learning. The first contribution was to introduce the Byzantine offline learning to train the historical spectrum sensing data. The second contribution was to propose a attacker-identification algorithm, based on Bayesian online learning, that is able to detect attackers and eliminate their influence on CSS.

REFERENCES

[1] A. S. Rawat, P. Anand, H. Chen, and P. K. Varshney, "Collaborative spectrum sensing in the presence of Byzantine attacks in cognitive radio networks," *IEEE Trans. Signal Process.*, vol. 59, no. 2, pp. 774–786, Feb. 2011.

[2] S. Haykin, "Cognitive radio: Brain-empowered wireless communications," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 2, pp. 201–220, Feb. 2005.

[3] T. Yucek and H. Arslan, "A survey of spectrum sensing algorithms for cognitive radio applications," *IEEE Commun. Surveys Tuts.*, vol. 11, no. 1, pp. 116–130, 1st Quart., 2009.

[4] L. Zhang, G. Ding, Q. Wu, Y. Zou, Z. Han, and J. Wang, "Byzantine attack and defense in cognitive radio networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 3, pp. 1342–1363, 3rd Quart., 2015.

[5] L. Zhang, G. Ding, Q. Wu, and F. Song, "Defending against byzantine attack in cooperative spectrum sensing: Defense reference and performance analysis," *IEEE Access*, vol. 4, pp. 4011–4024, Aug. 2016.

[6] A. Vempaty, L. Tong, and P. K. Varshney, "Distributed inference with Byzantine data: State-of-the-art review on data falsification attacks," *IEEE Signal Process. Mag.*, vol. 30, no. 5, pp. 65–75, Sep. 2013.

[7] J. Mitola and G. Q. Maguire, Jr., "Cognitive radio: Making software radios more personal," *IEEE Pers. Commun.*, vol. 6, no. 4, pp. 13–18, Apr. 1999.

[8] G. Ding *et al.*, "Robust spectrum sensing with crowd sensors," *IEEE Trans. Commun.*, vol. 62, no. 9, pp. 3129–3143, Sep. 2014.

[9] H. Li and Z. Han, "Catch me if you can: An abnormality detection approach for collaborative spectrum sensing in cognitive radio networks," *IEEE Trans. Wireless Commun.*, vol. 9, no. 11, pp. 3554–3565, Nov. 2010.

[10] Z. Qin, Q. Li, and G. Hsieh, "Defending against cooperative attacks in cooperative spectrum sensing," *IEEE Trans. Wireless Commun.*, vol. 12, no. 6, pp. 2680–2687, Jun. 2013.

[11] L. Zhang, Q. Wu, G. Ding, S. Feng, and J. Wang, "Performance analysis of probabilistic soft SSDF attack in cooperative spectrum sensing," *EURASIP J. Adv. Signal Process.*, no. 1, pp. 1–81, May 2014.

[12] X. He, H. Dai, and P. Ning, "HMM-based malicious user detection for robust collaborative spectrum sensing," *IEEE J. Sel. Areas. Commun.*, vol. 31, no. 11, pp. 2196–2208, Nov. 2013.

[13] R. Chen, J.-M. Park, and K. Bian, "Robust distributed spectrum sensing in cognitive radio networks," in *Proc. IEEE INFOCOM*, Phoenix, AZ, USA, Apr. 2008, pp. 13–18.

[14] X. He, H. Dai, and P. Ning, "A Byzantine attack defender in cognitive radio networks: The conditional frequency check," *IEEE Trans. Wireless Commun.*, vol. 12, no. 5, pp. 2512–2523, May 2013.

[15] B. Kailkhura, Y. S. Han, S. Brahma, and P. K. Varshney, "Distributed Bayesian detection in the presence of Byzantine data," *IEEE Trans. Signal Process*, vol. 63, no. 19, pp. 5250–5263, Oct. 2015.

[16] B. Kailkhura, S. Brahma, B. Dulek, Y. S. Han, and P. K. Varshney, "Distributed detection in tree networks: Byzantines and mitigation techniques," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 7, pp. 1499–1512, Jul. 2015.

[17] T. Zhao and Y. Zhao, "A new cooperative detection technique with malicious user suppression," in *Proc. ICC*, Dresden, Germany, Jun. 2009, pp. 14–18.

[18] M. Jo, L. Han, D. Kim, and H. P. In, "Selfish attacks and detection in cognitive radio ad-hoc networks," *IEEE Network.*, vol. 27, no. 3, pp. 46–50, May/Jun. 2013.

[19] P. Kaligineedi, M. Khabbazian, and V. K. Bhargava, "Malicious user detection in a cognitive radio cooperative sensing system," *IEEE Trans. Wireless Commun.*, vol. 9, no. 8, pp. 2488–2497, Aug. 2010.

[20] S. Althunibat, B. J. Denise, and F. Granelli, "Identification and punishment policies for spectrum sensing data falsification attackers using delivery-based assessment," *IEEE Trans. Veh. Technol.*, vol. 65, no. 9, pp. 7308–7321, Sep. 2016.

[21] G. Nie, G. Ding, and L. Zhang, "Byzantine attacker identification in collaborative spectrum sensing via Bayesian learning," in *Proc. EAI Int. Conf. Adv. Hybrid Inf. Process. China (ADHIP)*, Harbin, China, Jul. 2017, pp. 1–5.

[22] M. Opper, "A Bayesian approach to on-line learning," in *On-Line Learning in Neural Networks*, D. Saad, Ed. Cambridge, U.K.: Cambridge Univ. Press, 1998, pp. 363–378.

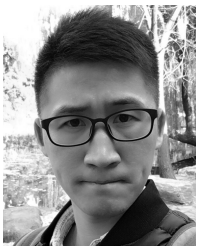
[23] S. Solla and O. Winther, "Optimal perceptron learning: An on-line Bayesian approach," in *On-Line Learning in Neural Networks*, D. Saad, Ed. Cambridge, U.K.: Cambridge Univ. Press, 1998, pp. 379–398.

[24] Q. Wu, G. Ding, J. Wang, and Y.-D. Yao, "Spatial-temporal opportunity detection for spectrum-heterogeneous cognitive radio networks: Two-dimensional sensing," *IEEE Trans. Wireless Commun*, vol. 12, no. 2, pp. 516–526, Feb. 2013.

[25] H. Urkowitz, "Energy detection of unknown deterministic signals," *Proc. IEEE*, vol. 55, no. 4, pp. 523–531, Apr. 1967.

[26] W. Zhang, R. K. Mallik, and K. B. Letaief, "Optimization of cooperative spectrum sensing with energy detection in cognitive radio networks," *IEEE Trans. Wireless Commun.*, vol. 8, no. 12, pp. 5761–5766, Dec. 2009.

[27] Z. Chair and P. K. Varshney, "Optimal data fusion in multiple sensor detection systems," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-22, no. 1, pp. 98–101, Jan. 1986.



GUANGMING NIE received the B.S. degree in electronic engineering from the Changsha University of Science and Technology, Changsha, China, in 2015. He is currently pursuing the M.S. degree with the College of Communications Engineering, PLA University of Science and Technology. His research interests include data security, wireless communications, and cognitive radio networks.

GUORU DING (S'10–M'14–SM'16) received the B.S. degree (Hons.) in electrical engineering from Xidian University, Xi'an, China, in 2008, and the Ph.D. (Hons.) degree in communications and information systems from the College of Communications Engineering, Nanjing, China, in 2014. Since 2014, he has been an Assistant Professor with the College of Communications Engineering and a Research Fellow with the National High Frequency Communications Research Center of China. Since 2015, he has been a Post-doctoral Research Associate with the National Mobile Communications Research Laboratory, Southeast University, Nanjing. His research interests include cognitive radio networks, massive MIMO, machine learning, and big data analytics over wireless networks.

Dr. Ding was a recipient of the best paper awards from EAI MLICOM 2016, the IEEE Vehicular Technology Conference (VTC) 2014-Fall, and IEEE WCSP 2009, the Alexander von Humboldt Fellowship in 2017, and the Excellent Doctoral Thesis Award of China Institute of Communications in 2016. He has served as a Guest Editor of the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS (Special issue on spectrum sharing and aggregation in future wireless networks). He is currently an Associate Editor of the *Journal of Communications and Information Networks*, the *KSII Transactions on Internet and Information Systems*, and the *AEU-International Journal of Electronics and Communications*. He has acted as a Technical Program Committees Member for a number of international conferences, including the IEEE Global Communications Conference, the IEEE International Conference on Communications, and VTC. He is a Voting Member of the IEEE 1900.6 Standard Association Working Group.

LINYUAN ZHANG received the B.S. degree (Hons.) in electronic engineering from Inner Mongolia University, Hohhot, China, in 2012. He is currently pursuing the M.S. degree in communications and information system with the College of Communications Engineering, PLA University of Science and Technology. His research interests are wireless communications and cognitive radio networks.

QIHUI WU received the B.S. degree in communications engineering, the M.S. degree, and the Ph.D. degree in communications and information systems from the Institute of Communications Engineering, Nanjing, China, in 1994, 1997, and 2000, respectively. From 2003 to 2005, he was a Post-doctoral Research Associate, Southeast University, Nanjing, China. From 2005 to 2007, he was an Associate Professor with the Institute of Communications Engineering, PLA University of Science and Technology, Nanjing, China, where he served as a Full Professor from 2008 to 2016. Since 2016, he has been a Full Professor with the College of Electronic and Information Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, China. In 2011, he was an Advanced Visiting Scholar with the Stevens Institute of Technology, Hoboken, USA.

His current research interests span the areas of wireless communications and statistical signal processing, with an emphasis on the system design of software-defined radio, cognitive radio, and smart radio.