Int J Wireless Inf Networks (2017) 24:344–355
DOI 10.1007/s10776-017-0363-2

Primary User Emulation Detection Algorithm Based on Distributed Sensor Networks

Di Pu[1] · **Bengi Aygun**[1] · **Alexander M Wyglinski**[1]

Received: 8 February 2016 / Accepted: 30 June 2017 / Published online: 12 July 2017

Abstract In this paper, we propose a novel primary user emulation (PUE) detection approach which employs a distributed sensor network, where each sensor node operates as an independent PUE detector. Distributed nodes collaborate in order to obtain the final detection results for the whole network. A voting algorithm is used to improve the performance of energy detection, while the classification is conducted by the nearest node in order to improve the efficiency of the detector. As a result of voting, if a potential primary user exists, then the features of the unknown user is compared with entries from the database in order to obtain a solid detection match. An artificial neural network (ANN) is used for the classification of an unknown user. To assess the accuracy of the detection result, we implement a reliability check at the output of ANN. The proposed algorithm is validated via computer simulations as well as by experimental hardware implementations using the Universal Software Radio Peripheral (USRP) software-defined radio (SDR) platform. The experiment results show that the distributed network detector detects the PUE 180–200%, depending on the number of primary users, faster than single node detector.

Keywords Cognitive radio network · Primary user emulation · Relational database · Frequency domain action recognition · Distributed sensor network

1 Introduction

Given the rapid growth of the wireless sector, the amount of available wireless spectrum to support a wide range of applications and services continues to decrease at a significant rate. This is partially due to the static, inflexible nature of wireless spectrum assignments defined by legacy regulatory guidelines and processes. With most wireless spectrum ranging between 0 and 3 GHz already allocated via static assignments to a range of governmental, corporate, and academic entities [1], and numerous instances exist where multiple spectrum assignments have been made for several frequency bands, this frequency assignment situation has resulted in fierce competition for the use and access of wireless spectrum. This is especially true in frequency bands located below 3 GHz, which is considered to be "prime" spectral real estate. Conversely, a large portion of the assigned spectrum has been observed during several spectrum measurement campaigns [2, 3] to be sparsely and sporadically utilized. In particular, spectrum occupancy by licensed transmissions are often concentrated across specific frequency ranges while a significant amount of the spectrum remains either underutilized or completely unoccupied. To remedy this spectrum scarcity issue, *dynamic spectrum access* (DSA) has been proposed as a solution, where wireless access is provided to unlicensed applications and users (i.e., secondary users) by allowing them to temporarily borrow unoccupied licensed spectrum while simultaneously guaranteeing the rights of incumbent licensed users (i.e., primary users). These primary users (PUs) possess a substantially higher priority or legacy rights across their assigned portion of wireless spectrum. Nevertheless, the secondary users (SUs) are permitted to access this spectrum as long as they do not

✉ Bengi Aygun
baygun@wpi.edu

[1] Worcester Polytechnic Institute, Worcester, MA, USA

interfere with PUs, thus enabling efficient utilization of spectral resources [4–6].

One of the major technical challenges regarding spectrum sensing is the problem of accurately distinguishing between primary user signals and secondary user signals [7]. In cognitive radio networks, primary users possess priority access to the channel, while secondary users must always relinquish access to the channel to the primary user and ensure that no interference is generated. Consequently, if a primary user begins to transmit across a frequency band occupied by a secondary user, the secondary user is required to leave that specific spectral band immediately. Conversely, when there is no primary user activity present within a frequency range, all the secondary users are permitted some level of access to the unoccupied frequency channel. Based on this principle, there exists the potential for malicious secondary users to mimic the characteristics of the primary users in order to gain priority access to the wireless channels occupied by other secondary users. This scenario is referred to in the literature as a *primary user emulation* (PUE) attack [8–10].

In order to overcome the problem of PUE, the FCC currently employs a centralized control approach [11, 12]. However, there are several issues related to this approach. First, in some cases this type of centralized control is not feasible, such as an emergency/disaster relief situation [13–15], or a military application [16–18], where there is no Internet or base station infrastructure available. Second, this type of centralized control may potentially be inefficient. The request and acknowledgement process could incur significant overhead to the network, and the detection process may suffer from latency issues before it receives an acknowledgement from the central. In [19], the voting algorithm to detect PUE is introduced. However, this approach does not have any reliability check. In [20], genetic algorithms are applied to VSUs in the presence of PUEA. Although the genetic algorithms are robust mechanisms, long convergence time makes it unpractical. In [21–23], the physical layer characteristics are used to detect PUE.

Given the published solutions currently available in the open literature, there still exists several technical challenges associated with enabling primary user emulation detection in cognitive radio networks, namely:

- Simple energy detector-base schemes possess a significant probability of missed detection.
- Signature-based detection and most of the feature detection methods require special hardware and software.
- Analytical model-based detection approach works well for a specific network model, but it may not work well for the other models.
- Localization-based detection can only be employed for stationary primary transmitters with known coordinates.

To resolve these issues, in this paper we propose a distributed PUE detection algorithm, where each node detects the energy levels across the spectrum. All nodes vote based on their spectrum sensing measurements due to the possible existence of PUE [24, 25]. In case of a potential PU being present, the data from the unknown user is collected from the nearest node, and this data is compared with the primary user database. If its feature matches with the database, an artificial neural network analyzes the temporal-spectral actions of unknown user. In the meantime, the reliability of this analysis is also assessed. Finally, the proposed approach decides whether the unknown user is an authentic PU or a malicious PUE. The proposed detector possesses the following novel contributions:

- The distributed nodes in the network collaborate in order to detect the unknown user but only the node nearest to the unknown user is employed to collect the data. Using this approach, we show later on in the results section that the detection duration is approximately 200% faster than the single node detection.
- The proposed approach does not need any special hardware or software required in order to operate. In addition, it can be employed without significant structural and functional modifications.
- The detection reliability increases by the increasing number of distributed nodes in the network.
- The proposed approach is robust in the presence of noise since the voting is performed across many different nodes.

The rest of this paper is organized as follows: In Sect. 2, we present the system model used in this work. In addition, we denote the hypothesis testing framework used for performing the detection process. In Sect. 3, the proposed algorithm is derived. In Sect. 4 we obtain the probability of detection and false alarm to evaluate the performance of proposed algorithm. We present the both computer and hardware experiment results in Sect. 5. Finally, we conclude the paper with several comments in Sect. 6.

2 System Model

Most PUE detection approaches assume there is a single-node PUE detector within the network. However, in order to improve the efficiency and accuracy of the detection process, we can alternatively employ a distributed sensor network for the purposes of detection, as shown in Fig. 1, where each sensor node works as an independent PUE

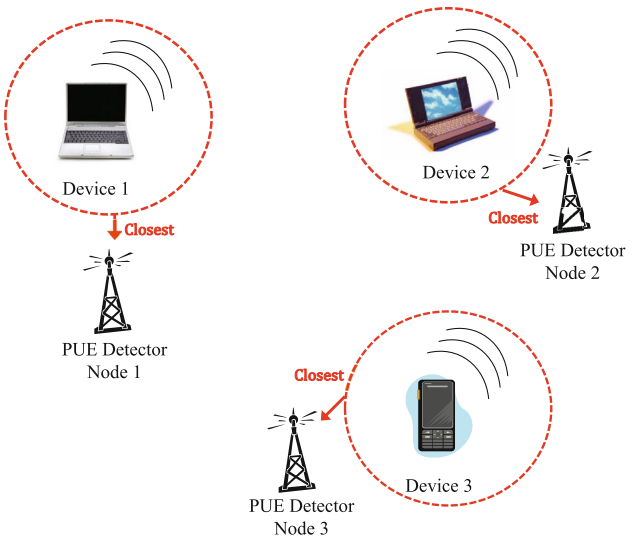


Fig. 1 A contention based dynamic spectrum access network that employs a three-node distributed sensor network for performing PUE detector, where each sensor node works as an independent PUE detector. For an unknown user in this network, all sensor nodes perform the initial detection of the device, but the node closest to the device makes the final decision result

detector. For an unknown user in this network, each sensor node makes its own decision concerning whether it is a primary user emulator or not. Based upon this approach, a final detection result can be made.

Each node of this sensor network can employ one of the approaches proposed in [9, 10]. However, the emphasis of this work is how these nodes collaborate with each order in order to obtain the final detection results for the whole network.

We consider a cognitive radio network as shown in Fig. 2. All the users, including the primary users, primary user emulators, and secondary users, as well as the PUE detectors, are distributed in a circular grid. In order to avoid interference, we assume at each time that there is only one user transmitting in this network.

We denote $x(t)$ as the transmitted signal. If it is an authentic PU signal, the notation of the signal is $x(t) = s(t)$. Otherwise, it is the PUE signal and the notation

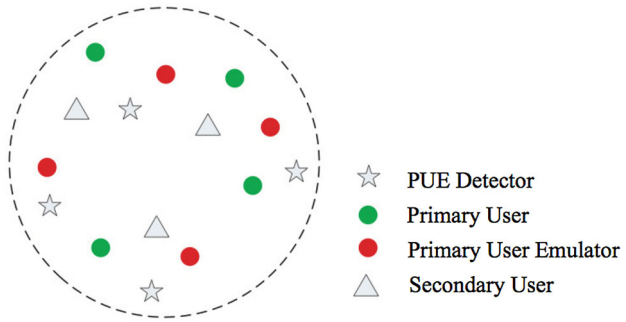


Fig. 2 A cognitive radio network in a circular grid

is $x(t) = s'(t)$. Since the PUE signal is very similar to a PU signal, we assume that both the $s(t)$ and $s'(t)$ signals are independently and identically distributed (iid) random processes with mean zero and variance σ_s^2, namely:

$$s(t), s'(t) \sim \mathcal{N}\left(0, \sigma_s^2\right), \tag{1}$$

where $\mathcal{N}(\cdot)$ denotes the normal distribution. Since the secondary users have a significant lower transmitted power than the primary users, we assume $x(t) = 0$ when the SU is transmitting.

Suppose we let $h_i(t)$ and $n_i(t)$ denote the impulse response and the noise of the channel between the transmitted signal and the ith PUE detector. Furthermore, we assume that the channel is a slow flat fading channel during the observation process. Such that $h_i(t)$ becomes a constant gain, i.e., h_i, and $n_i(t)$ is additive white Gaussian noise (AWGN) with mean zero and variance σ_n^2, namely:

$$n_i(t) \sim \mathcal{N}\left(0, \sigma_n^2\right). \tag{2}$$

Using these models, there are three possible received signals for the ith ($1 \leq i \leq M$) PUE detector, namely [26]:

$$y_i(t) = \begin{cases} n_i(t) & \rightarrow \text{SU}, \\ h_i \times s(t) + n_i(t) & \rightarrow \text{PU}, \\ h_i \times s'(t) + n_i(t) & \rightarrow \text{PUE}, \end{cases} \tag{3}$$

where $y_i(t)$ is the received signal at the ith PUE detector. The PUE detection algorithm presented in Sect. 3 will differentiate between these three cases at each detector, and then combine their results into a single final decision.

2.1 Hypothesis Testing

Spectrum sensing is employed for the purpose of identifying unoccupied licensed spectrum, which is equivalent to detecting the frequency locations of the primary user signals. Therefore, spectrum sensing process can be interpreted as a signal detection problem. Since most signal detection problems can be formulated in the framework of an M-ary hypothesis test, where we have an observation (possibly a vector or function) upon which we wish to decide among M possible statistical situations describing the observations [27]. According to this criterion, the spectrum sensing performs a binary hypothesis test in order to decide whether or not there are primary signals in a particular channel. The two hypotheses are denoted as follows:

$$\begin{aligned} &\mathcal{H}_0 : \text{no primary signals}, \\ &\mathcal{H}_1 : \text{primary signals exist}, \end{aligned} \tag{4}$$

where $\mathcal{H}_0$ is usually referred to as a null hypothesis, and $\mathcal{H}_1$ is usually called the alternative hypothesis. On the other hand, for the alternative hypothesis, the received signal

would be the superposition of the noise and the primary user signals. Thus, the two hypotheses in Eq. (4) can be represented as:

$$\begin{aligned} \mathcal{H}_0 &: x_k = n_k, \\ \mathcal{H}_1 &: x_k = s_k + n_k, \end{aligned} \tag{5}$$

for $k = 1, \ldots, N$, where N is the number of received signals, x_k is the received signal, n_k is the noise in the RF environment, and s_k is the primary signal. Consequently, the spectrum sensing process can be considered as a detection problem, such that based on the observation x, we need to decide among two possible statistical situations describing the observation, which can be expressed as:

$$\delta(x) = \begin{cases} 1 & x \in \Gamma_1, \\ 0 & x \in \Gamma_1^c. \end{cases} \tag{6}$$

When the observation x falls inside the region Γ_1, we will choose $\mathcal{H}_1$. However, if the observation falls outside the region Γ_1, we will choose $\mathcal{H}_0$. Therefore, Eq. (6) is known as a *decision rule*, which is a function that maps an observation to an appropriate hypothesis [27]. In the context of spectrum sensing, different spectral detectors and classifiers are actually the implementations of different decision rules. In this paper, we consider energy detection as a decision rule.

Regardless of the precise signal model or detector used, sensing errors are inevitable due to additive noise, limited observations, and the inherent randomness of the observed data [28]. By testing $\mathcal{H}_0$ versus $\mathcal{H}_1$ in Eq. (4), there are two types of errors that can be made, namely $\mathcal{H}_0$ can be falsely rejected or $\mathcal{H}_1$ can be falsely rejected [27]. In the first hypothesis, there are actually no primary signals in the channel, but the testing detects an occupied channel, so this type of error is called a *false alarm* or *Type I error*. In the second hypothesis, there actually exist primary signals in the channel but the testing detects only a vacant channel. Thus, we refer to this type of error as a *missed detection* or *Type II error*. Consequently, a false alarm may lead to a potentially wasted opportunity for the SU to transmit while a missed detection could potentially lead to a collision with the PU [28].

Given these two types of errors, the performance of a detector can be characterized by two parameters, namely, the *probability of false alarm* (P_F), and the *probability of missed detection* (P_M) [29], which corresponds to Type I and Type II errors, and can be defined as:

$$P_F = P\{\text{Decide}\,\mathcal{H}_1|\mathcal{H}_0\}, \tag{7}$$

$$P_M = P\{\text{Decide}\,\mathcal{H}_0|\mathcal{H}_1\}. \tag{8}$$

Note that based on P_M, another frequently used parameter is the *probability of detection*, which can be derived as follows:

$$P_D = 1 - P_M = P\{\text{Decide}\,\mathcal{H}_1|\mathcal{H}_1\}, \tag{9}$$

which characterizes the detector's ability to identify the primary signals in the channel, where P_D is usually referred to as the power of the detector.

As for the detectors, we would like their probability of false alarm to be as low as possible, and at the same time their probability of detection to be as high as possible. However, in a real-world situation, this is not achievable because these two parameters are constraining each other. To show their relationship, a plot called the *receiver operating characteristic* (ROC) is usually employed [30], as shown in Fig. 3, where its x-axis is the probability of false alarm and its y-axis is the probability of detection. From this plot, we observe that as P_D increases, the P_F is also increasing. There does not exist such an optimal point that reaches the highest P_D and the lowest P_F. Therefore, the detection problem is also a trade off, which depends on how the Type I and Type II errors should be balanced.

3 Proposed PUE Detection Algorithm

The proposed PUE detection algorithm in this work uses a voting process across all nodes within the sensor network. A flow diagram of the algorithm is provided in Fig. 4. Although we show the database approach in the flow chart, the nearest node can employ any of the three approaches proposed in [9, 10] when working as a classifier in the second step. According to the system model, our proposed approach makes the following assumptions: (i) All the users, including the malicious users and primary users, are

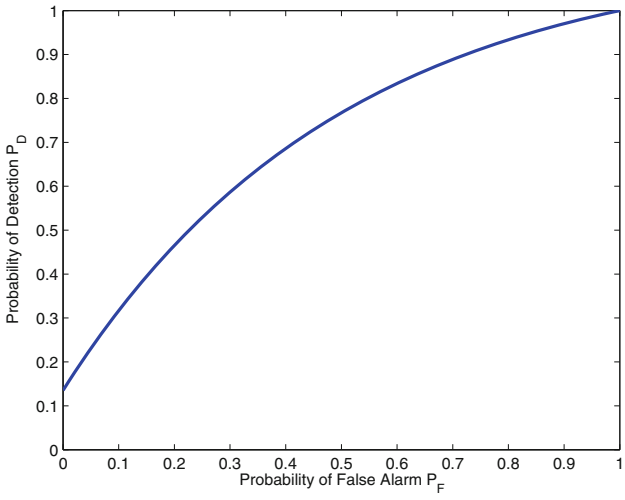


Fig. 3 A typical receiver operating characteristic (ROC), where the x-axis is the probability of false alarm (P_F), and the y-axis is the probability of detection (P_D)

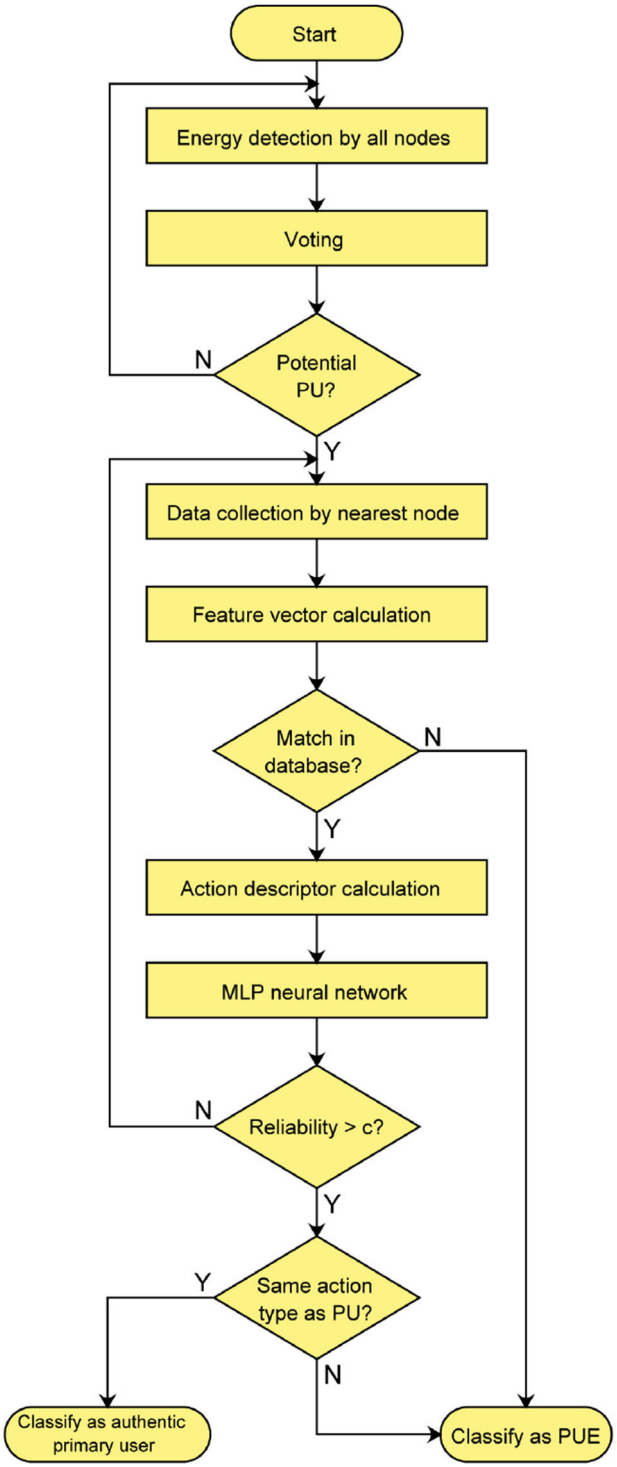


Fig. 4 Proposed PUE detection algorithm based on distributed sensor network

located within the same frequency band; (ii) For each period of time, there is only one user transmitting.

The algorithm begins with an initialization stage, which uses energy detection in order to determine the frequency location of the potential primary user. For each given interval, all the detector nodes scan the same frequency bin and try to differentiate the following two cases of the received signal:

$$y_i(t) = \begin{cases} n_i(t) & \rightarrow \text{ SU}, \\ h_i \times x(t) + n_i(t) & \rightarrow \text{ PU \& PUE}, \end{cases} \tag{10}$$

where $y_i(t)$ is the received signal at the ith detector, $x(t) = s(t)$ or $s'(t)$.

In this step, each sensor node will make a detection decision concerning whether the received signal belongs to a potential PU or not. Suppose there are M sensor nodes in the network, and the detection result is either "1", i.e. (potential PU present) or "0", i.e. (potential SU present). For an unknown received signal, the result is:

$$r = \begin{cases} 1 & \sum_{i=1}^{M} r_i \geq \frac{M}{2}, \\ 0 & \text{otherwise}, \end{cases} \tag{11}$$

where r_i is the detection result from the ith sensor node.

In other words, if the majority of the sensor nodes decide this is a potential primary user, the algorithm will continue to the second step and this signal will be recorded by its nearest node. After a certain period of time T, this observation process is terminated and the recorded signals are passed on to the classifier.

This proposed algorithm features the energy detection by all the nodes and data collection by only the nearest node. Thus, it eliminates substantial amount of the overhead across a long sensing time period caused by energy detection and high computation levels due to feature calculations. The collected data is used for calculating the feature vector, which defines the temporal-spectral action of the user. The feature vector of primary user is compared with the data in the database system, which stores the feature vectors of the primary users. In most applications, primary users possess routine wireless transmissions, so they have a limited number of feature vectors, which means the resulting database is stable and limited in size. In case an unknown user feature vector has a match entity in the database, this approach will continue to double check its action in the frequency domain using artificial neural network. Otherwise, this unknown user will be classified as a PUE. Binary Multilayer Perception (MLP) classifies the users to detect whether it is PUE or PU. The hidden layers calculates the output based on the weights of input values. The number of layers in hidden levels and the number of hidden perceptrons in one layer are predefined by the designer [31, 32]. In this work, we used MATLAB MLP toolbox to detect PUEs [33]. Our approach operates on intercepted signals and analyzes it in the frequency domain over a time interval. Besides the benefits of our previous approach, our new approach takes the stability of primary

users into account and creates a database system such that it is can reduce the level of computational complexity [10]. Artificial neural networks are used to classify the signal based on the temporal-spectral action. If the reliability of the testing result is less than a constant number c specified at the beginning, we need to collect some new received data and run these procedures again. Otherwise, the neural network will output the classification result. Since the temporal-spectral action type of primary users is known, we can readily identify whether the observed signal is from a real primary user or a malicious secondary user.

4 Probability of Detection and False Alarm

In this paper, we study how the sensor network will impact the energy detector performance in terms of probability of the false alarm and probability of detection. For each detector node, the probability of false alarm and probability of detection will be derived. Since both n_k and x_k are iid normal random variables, y_k has the following distribution:

$$y_k \sim \begin{cases} \mathcal{N}(0, \sigma_0^2) & \mathcal{H}_0, \\ \mathcal{N}(0, \sigma_1^2) & \mathcal{H}_1, \end{cases} \tag{12}$$

where $\sigma_0^2 = \sigma_n^2$ and $\sigma_0^2 = h^2\sigma_s^2 + \sigma_n^2$. Consequently, a decision statistic for energy detector can be defined as:

$$Y = \sum_{k=1}^{N} |y_k|^2, \tag{13}$$

where x_k and N follow the definitions in Eq. (5). Under both hypotheses, the decision statistic $\mathcal{Y}$ is the sum of the squares of N mutually independent normal random variables, where $\mathcal{Y}$ has the central chi-square distribution with $2N$ degrees of freedom.

Given the decision statistic Y and a threshold T, the performance of this energy detector can be characterized by two parameters, namely, the probability of false alarm (P_F), and the probability of detection (P_D), which can be defined as:

$$\begin{aligned} P_F &= P(\text{Decide } \mathcal{H}_1|\mathcal{H}_0) = \mathrm{P}(\mathrm{Y} > \mathrm{T}|\mathcal{H}_0) \\ &= P\big(Y > T|Y \sim \mathcal{X}^2_{2N}\sigma_0^2\big). \end{aligned} \tag{14}$$

$$\begin{aligned} P_D &= P(\text{Decide } \mathcal{H}_1|\mathcal{H}_1) = \mathrm{P}(\mathrm{Y} > \mathrm{T}|\mathcal{H}_1) \\ &= P\big(Y > T|Y \sim \mathcal{X}^2_{2N}\sigma_1^2\big). \end{aligned} \tag{15}$$

We already know that the cumulative distribution function (CDF) of a standard central chi-square distribution is given by:

$$F(x;k) = \frac{\gamma(k/2, x/2)}{\Gamma(k/2)}, \tag{16}$$

where k is the degree of freedom, $\gamma(\cdot,\cdot)$ is the lower incomplete gamma functions, and $\Gamma(\cdot)$ is the ordinary gamma function. Since Eqs. (14) and (15) are complement of the CDF defined in Eq. (16), they can be obtained by:

$$P_F = 1 - F\left(\frac{T}{\sigma_0^2}; 2N\right) = \frac{\Gamma\left(N, \frac{T}{2\sigma_0^2}\right)}{\Gamma(N)} \tag{17}$$

$$P_D = 1 - F\left(\frac{T}{\sigma_1^2}; 2N\right) = \frac{\Gamma\left(N, \frac{T}{2\sigma_1^2}\right)}{\Gamma(N)} \tag{18}$$

where $\Gamma(\cdot,\cdot)$ is the upper incomplete gamma function and $\Gamma(\cdot)$ is the ordinary gamma function. Since we use voting to determine the result of energy detection, the overall probability of false alarm and probability of detection can be calculated using the law of total probability:

$$Q_F = \sum_{i=\frac{M}{2}}^{M} \binom{M}{i} P_F^i (1 - P_F)^{M-i}, \tag{19}$$

and

$$Q_D = \sum_{i=\frac{M}{2}}^{M} \binom{M}{i} P_D^i (1 - P_D)^{M-i}. \tag{20}$$

5 Experimental Setup and Results

In this section, two experiments are conducted in order to validate the performance of the classifier conducted by the nearest node. The first experiment uses a computer simulation based on Simulink, while the second experiment is based on a hardware implementation using the Universal Software Radio Peripheral (USRP) software-defined radio (SDR) platform [34]. The classifier is tested in two aspects: accuracy and efficiency. Note that these tests possess with an emphasis on the impact of distance, i.e., nearest node.

5.1 Computer Simulation

In this section, the ROC curve analysis is shown for different number of users. Additionally, a Simulink model is constructed in order to collect the FFT plot of a user. Before presenting the results, the path-loss modeling will be introduced since path-loss block is used in the experiments.

5.1.1 Path-loss Modeling

In most environment, it is observed that the radio signal strength falls as some power α of the distance, called the power-distance gradient or path-loss gradient. Depending on the radio frequency, there are additional losses, and in general the relationship between the transmitted power P_t and the received power P_r in free space is given by:

$$\frac{P_r}{P_t} = G_t G_r \left(\frac{\lambda}{4\pi d}\right)^2, \tag{21}$$

where G_t and G_r are the transmitter and receiver antenna gains, λ is the wavelength of the carrier, and d is the distance between the transmitter and receiver. Since all the detector nodes are equipped with the same antenna, it turns out that G_r is the same. According to our assumptions, for each period of time, there is only one user transmitting, which means that G_t and λ are the same. Therefore, given the transmitted power, the only variable for the received power is d. We can rewrite Eq. (21) in decibels (dB) as:

$$10\log(P_r) = 10\log(P_0) - 20\log(d), \tag{22}$$

where P_0 is the received power at the first meter ($d = 1$), which applies to all the detector nodes. Therefore, there is a 20 dB per decade loss in signal strength as a function of distance in free space. It is known that the performance of the classifier is highly related to the signal-to-noise ratio (SNR) on the receiver, namely, higher SNR value yields better classification performance. Broadly speaking, SNR is the ratio of the average signal power to the average noise power:

$$SNR = \frac{P_r}{P_{noise}}. \tag{23}$$

Since the average noise power P_{noise} is fixed for all the detector nodes, in order to get a higher SNR value, a larger P_r is required. According to Eq. (22), a minimum distance d will lead to a maximum P_r, thus the nearest node is picked up to perform the classification.

5.1.2 ROC Curve Analysis

This section provides the numerical results of the proposed PUE detector. The results are based on Eqs. (19) and (20), presented in terms of the ROC curves (i.e., Q_D versus Q_F) for an AWGN fading channel.

For each sensor node, P_F and P_D are calculated. Since Eqs. (17) and (18) are the complement of the binomial cumulative distribution function [35], there is a useful MATLAB function `binocdf` that we can use in to obtain numerical results.[1]

In order to have a quick evaluation of the sensor network, we perform the following calculation, where $P_F = 0.1$ and $P_D = 0.8$. Assuming we use a sensor network of 4 nodes, then:

$$Q_D = 1 - \texttt{binocdf}(2, 4, 0.8) = 0.82 > P_D, \tag{24}$$

$$Q_F = 1 - \texttt{binocdf}(2, 4, 0.1) = 0.0037 < P_F, \tag{25}$$

which means that by employing the sensor network, we not only improve the overall probability of detection, but also decrease the overall probability of false alarm.

Based on the ROC curves of single node detector, the ROC curves of the distributed sensor network can be obtained by changing the number of sensor nodes M. The ROC curves in Fig. 5a are plotted by changing M from 4 to 10 when $N = 5$, $SNR = 5$ dBm, and the ROC curves in Fig 5b are plotted by changing M from 4 to 10 when $N = 5$, SNR=3 dBm. In these two figures, the ROC curves of the single node detector ($M = 1$) are also provided for reference.

Based on Fig. 5, we can come to the following conclusions:

- Employing the distributed sensor network is an effective way of improving the performance of the energy detection if each sensor node of this network has a reliable performance, as shown in Fig. 5a.
- If the sensor node does not have a good P_D, the overall performance of the sensor network can be even worse. For example, in Fig. 5b, when M=4, the distributed sensor network has a lower P_D than the single node detector in the area of $Q_F < 0.3$. Therefore, we need to make sure that each sensor node in this network has an acceptable performance.
- Given a probability of false alarm, the larger number of sensor nodes M will yield higher overall probability of detection.

5.1.3 Simulink Experiments

If there are N different users, then we need to have N different parameter settings for this model. Figure 6 shows the structure of this model.

In this model, there are three blocks on the transmitter side: a random number generator, a modulator, and a pulse-shaping filter. By picking up the different modulation type and filter type, or by setting the different parameters for these three blocks, we can express different users and get different FFT plots. Then, a free space path loss block[2] and an AWGN channel block are applied to emulate the transmission environment. Specifically, the free space path loss block is employed to simulate the loss of signal power due to the distance between transmitter and receiver as expressed in Eq. (22). This block reduces the amplitude of the transmitted signal by an amount related to d. Then, the

[1] `Y = binocdf(X,N,P)` computes a binomial CDF at each of the values in X using the corresponding number of trials in N and probability of success for each trial in P [36].

[2] The free space path loss block belongs to the RF Impairments Library.

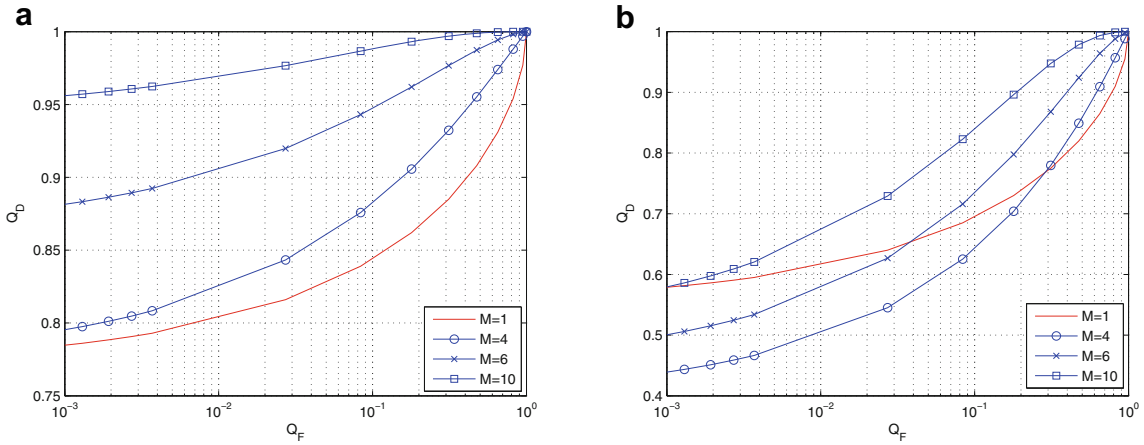


Fig. 5 Numerical results of the energy detector in terms of the ROC curves **a** ROC curves by varying the number of sensor nodes M from 4 to 10 given N=5, SNR=5. **b** ROC curves by varying the number of sensor nodes M from 4 to 10 given N=5, SNR=3

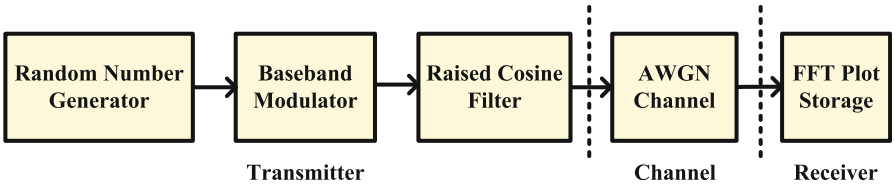


Fig. 6 The structure of the Simulink model used to collect the FFT plot of a user

AWGN channel block represents the noise level by setting the variance of the white Gaussian noise. In the end, a sink block is used on the receiver side to save the FFT plots. These plots are stored away in the workspace for post processing.

For each sensor node, a relational database is created to store the feature vectors of the primary users that are close to this node. In most cases, the primary users are stationary in the system, so the database is quite stable. Compared to the centralized approach, there is a significant advantage brought by the distributed sensor network. With the single node detection approach, all the feature vectors exist in one large database. However, with the distributed sensor network, those feature vectors will be divided into smaller databases. Therefore, it will greatly reduce the time to search the database. Although there are several widely used database management tools, such as MySQL, Oracle and Access, in order to facilitate the connection with the Simulink model, we build this database using MATLAB [37, 38]. We can use the `read` function to search the database and the `write` function to update the database, if there exists new primary user.

In this paper, an multi-layer perceptron (MLP) neural network with 256 input nodes, one layer of 6 hidden nodes, and one output node is employed [39]. $f(x) = \tanh(x)$ is selected as the activation function. For training, the back propagation algorithm is used with a fixed training constant of $\eta = 0.5$, and momentum constant $\zeta = 0.75$ [40]. The log-covariance descriptor vector is fed into the system of artificial neural networks, and the system outputs a classification result along with a reliability parameter. If the reliability parameter is larger than 0.75, the classification result is accepted.

The most important performance metric of a classifier is the percentage of correct classifications, which shows whether an approach is accurate. In most cases, we would like this percentage as high as possible. The first experiment will show how distance affects this percentage. In order to incorporate the variable of distance in a free space path loss block, we choose "Distance and Frequency" in the "Mode" field, and then specify the distance between transmitter and receiver. For a specific distance value, we can change the parameter settings on the transmitter side to generate different FFT plots such that we can repeat the experiment several times and average the resulting percentages. Figure 7 shows the percentage of correct classifications given by different distances d. Based on this figure, with distance values ranging from 1 to 10 m, the percentage of correct classifications drops dramatically from around 90–20%. More specifically, smaller distance values yield better algorithm performance in terms of successfully classifying primary signals and PUE signals. Therefore, in order to get the optimal classification performance, we need to pick up the nearest node to be the classifier.Moreover, if there is not a neighbor node whose reliability parameter is at least 75%, the algorithm does not take the neighbor node's input and trust only the individual transmitter's decision.

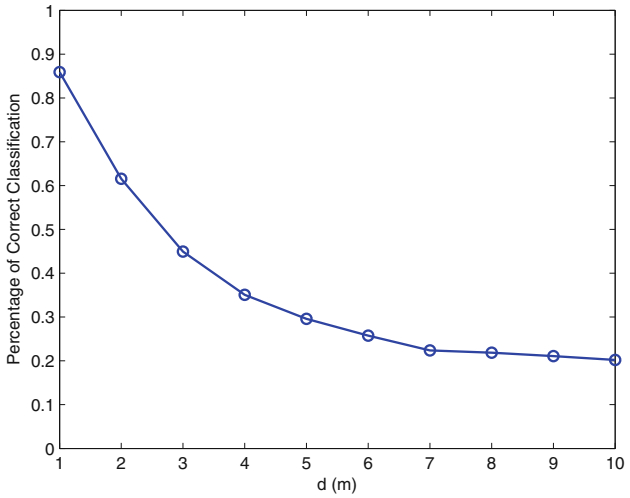


Fig. 7 The classification performance using the database-assisted approach in computer simulations. The x-axis represents distance value d, and the y-axis represents the percentage of correct classification. For the reliability check, all classifications with a reliability parameter less than 0.75 are ignored

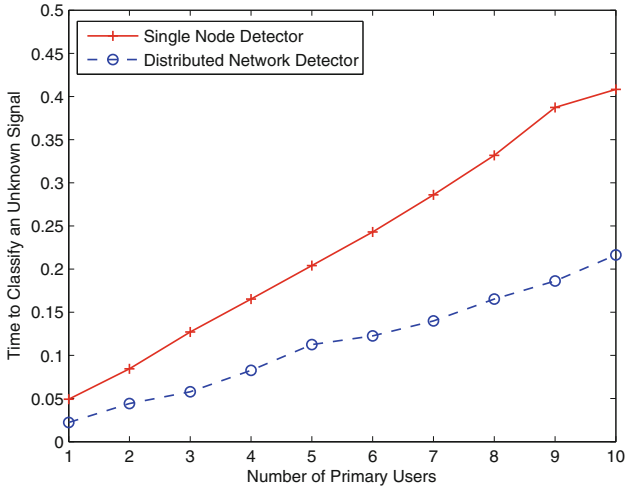


Fig. 8 Time in seconds to classify an unknown signal using the distributed network detector and the single node detector. The x-axis represents the number of primary users, and the y-axis represents the time to classify an unknown signal. For the reliability check, all classifications with a reliability parameter less than 0.75 are ignored

The second most important performance metric of a classifier is the time spent to classify the known signals, which shows whether an approach is efficient. In most cases, we would like this time as short as possible. The second experiment will show how distributed sensor network affects this time. As mentioned at the beginning of this section, one of the advantages of the nearest node classification is that all the feature vectors are divided into smaller databases. In many cases, we can get the result by just searching one or several small databases, and thus avoiding going through all the feature vectors. In Fig. 8, we compare the time to classify an unknown signal using the proposed distributed network detector and the single node detector. Assuming the distributed network detector consists of two sensor nodes, this time we do not include the training time of the artificial neural network because this process can be conducted offline with some previously known training signals. Once the artificial neural network has been trained, it only needs to be evaluated rather than both trained and evaluated with any newly intercepted signals.

It can be observed that for both approaches, the classification time is related to the number of primary users. When there are more primary users in the system, it costs more time to reach a conclusion. However, it should be noted that given a fixed number of primary users, it always takes less time for the nearest node approach to classify. According to the cost model introduced in [41], the database searching time is proportional to the number of tables to scan. Since each primary user has one corresponding table, when there are N primary users, there are N corresponding tables. Given an unknown user, the single detector approach has to scan all the tables in order to make the classification. However, the nearest node approach in this section only needs to scan the tables stored in the unknown user's nearest detector node to make the decision, thus saving a substantial amount of time.

5.1.4 *Software-Defined Radio Experiment*

The Simulink model in Fig. 9 was then used as a starting point for the design of a hardware implementation. For simplicity, a small scale distributed sensor network is constructed in this paper, which includes one unknown user as a transmitter and two sensor nodes as receivers. Based on Fig. 9, this was achieved by changing the AWGN channel block with a real-life fading channel, setting up the radios in different locations to represent the impact of the free space path loss block, and by using the Simulink SDRu blocks available in the Communications System Toolbox. Consequently, our resulting Simulink design that operates on the USRP SDR platform is shown in Fig. 9. In order to incorporate the USRP2 hardware into the existing Simulink model, two Simulink blocks called `SDRu Transmitter` and `SDRu Receiver` are used here as interfaces. These two blocks are developed by The MathWorks and have been available since the R2011a release of MATLAB [42]. With these two blocks, the USRP SDR platforms can be used in conjunction with the previous Simulink design environment.

The rest of the Simulink model remains the same, including the random number generator, baseband modulator, pulse shaping filter as shown in Fig. 9a, and FFT storage as shown in Fig. 9b. The next steps, including the database search as the feature vectors of the primary users are distributed in two databases.

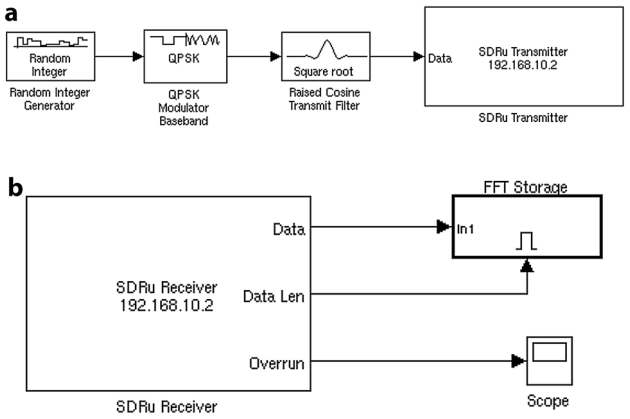


Fig. 9 The structure of hardware implementation framework. **a** The Simulink model for unknown user, which includes an SDRu Transmitter block. **b** The Simulink model for sensor node, which includes an SDRu Receiver block

Table 1 Software-Defined Radio Experimental Results

	Closer node (%)	Further node (%)
With check	90	47
Without check	85	43

Since there are two sensor nodes in the network, given an unknown user, there is one closer node, and one further node. In Table 1, the percentage of correct classifications of the two nodes is shown and compared with the hardware implementation. Similar to the results derived from computer simulations in Fig. 7, the closer node has a much better performance than the further node. Note that for the closer node, even without the reliability check, the percentage of correct detection can be as high as 85%, which means that the majority of the classification results are correct, so the proposed nearest node classification possesses the potential to be a viable component of the PUE detector operating under real world conditions.

In terms of execution and convergence times, we do not take the training time for the artificial neural network into account. When the FFT plot of an unknown user is collected, it takes 0.2 s for the nearest node to output the result, which is faster than that of the single detector approach. Considering both the efficiency and the performance, the nearest node classification approach proposed in this section is a good candidate for the real world implementation.

6 Conclusion

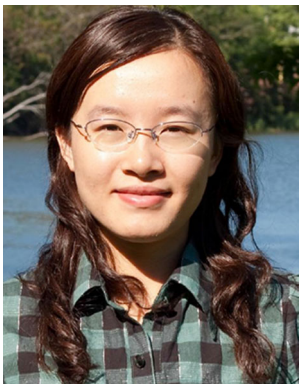
A novel algorithm for detecting non-intelligent primary user emulation attack based on distributed sensor network has been presented in this paper. This approach does not require any special hardware or software, and can be applied to mobile transmitters with unknown coordinates. Using USRP2 hardware experimentation, our work features an analysis in real-life channel with the effect of multipath fading and interference. Numerical results, computer simulations, and hardware implementations all demonstrate that the proposed approach possesses better performance related to the single detector in terms of the accuracy and efficiency. The future work of this approach will be focus on the node selection of the distributed sensor network.

Acknowledgements The authors would like to thank The MathWorks, Natick, MA, USA for their generous support of this research.

References

1. Federal Communications Commission (FCC), "Spectrum Inventory Table 137 MHz to 100 GHz," [Online]: http://www.fcc.gov/oet/info/database/spectrum/.
2. M. A. McHenry, P. A. Tenhula, D. McCloskey, D. A. Roberson, and C. S. Hood. Chicago spectrum occupancy measurements analysis and a long-term studies proposal. In *Proceedings of Workshop on Technology and Policy for Accessing Spectrum*, Boston, MA, August 2006.
3. S. Pagadarai and A. M. Wyglinski. A quantitative assessment of wireless spectrum measurements for dynamic spectrum access. In *Proceedings of the IEEE International Conference on Cognitive Radio Oriented Wireless Networks and Communications*, Hannover, Germany, June 2009.
4. D. Cabric, S. Mishra, D. Willkomm, R. Brodersen, and A. Wolisz. A cognitive radio approach for usage of virtual unlicensed spectrum. In *Proceedings of the 14th IST Mobile and Wireless Communications Summit*, June 2005.
5. Q. Zhao and B. M. Sadler, A survey of dynamic spectrum access, *IEEE Sginal Processing Magazine*, Vol. 24, No. 3, pp. 79–89, 2007.
6. Y. Chen and H. Oh, A survey of measurement-based spectrum occupancy modeling for cognitive radios, *IEEE Communications Surveys Tutorials*, Vol. 18, No. 1, pp. 848–859, 2016.
7. M. Ozger and O. Akan, On the utilization of spectrum opportunity in cognitive radio networks, *IEEE Communications Letters*, Vol. 20, No. 1, pp. 157–160, 2016.
8. Z. Jin, S. Anand, and K. P. Subbalakshmi. Detecting primary user emulation attacks in dynamic spectrum access networks. In *Proceedings of the IEEE International Conference on Communications*, Dresden, Germany, June 2009.
9. D. Pu, Y. Shi, A. V. Ilyashenko, and A. M. Wyglinski. Detecting primary user emulation attack in cognitive radio networks. In *Proceedings of the IEEE Global Telecommunications Conference*, December 2011.
10. D. Pu and A. M. Wyglinski. Primary user emulation detection using frequency domain action recognition. In *Proceedings of the 2011 IEEE Pacific Rim Conference on Communications, Computers, and Signal Processing*, August 2011.
11. Spectrum Bridge, "White Space Overview," [Online]: http://spectrumbridge.com/ProductsServices/WhiteSpacesSolutions/WhiteSpaceOverview.aspx.
12. X.-L. Huang, J. Wu, W. Li, Z. Zhang, F. Zhu and M. Wu, Historical spectrum sensing data mining for cognitive radio enabled vehicular ad-hoc networks, *IEEE Transactions on Dependable and Secure Computing*, Vol. 13, No. 1, pp. 59–70, 2016.

13. F. Li and K. Wu. Reliable, distributed and energy-efficient broadcasting in multi-hop mobile ad hoc networks. In *Proceedings of the 27th Annual IEEE Conference on Local Computer Networks*, Tampa, FL, November 2002.
14. R. Aldunate, S. F. Ochoa, F. Peña-Mora and M. Nussbaum, Robust mobile ad hoc space for collaboration to support disaster relief efforts involving critical physical infrastructure, *Journal of Computing in Civil Engineering*, Vol. 20, pp. 13–27, 2006.
15. Q. Liang, Ad hoc wireless network traffic self-similarity and forecasting, *IEEE Communications Letters*, Vol. 6, pp. 297–299, 2002.
16. K. Wongthavarawat and A. Ganz. IEEE 802.16 based last mile broadband wireless military networks with quality of service support. In *Proceedings of the IEEE Military Communications Conference*, Vol. 2, pp. 779–784, October 2003.
17. K. Jain, J. Padhye, V. N. Padmanabhan and L. Qiu, Impact of interference on multi-hop wireless network performance, *Wireless Networks*, Vol. 11, pp. 471–487, 2005.
18. M. J. Zieniewicz, C. Douglas, D. C. Wong and J. D. Flatt, The evolution of army wearable computers, *IEEE Pervasive Computing*, Vol. 1, pp. 30–40, 2002.
19. Z. Jin, S. Anand, and K. Subbalakshmi, Robust spectrum decision protocol against primary user emulation attacks in dynamic spectrum access networks. In *Proceedings of IEEE Global Telecommunications Conference*, Miami, FL, USA, December 2010.
20. D. Das and S. Das, Adaptive resource allocation scheme for cognitive radio vehicular ad-hoc network in the presence of primary user emulation attack, *IET Networks*, Vol. 6, No. 1, pp. 5–13, 2017.
21. N. Gao, X. Jing, H. Huang and J. Mu, Robust collaborative spectrum sensing using phy-layer fingerprints in mobile cognitive radio networks, *IEEE Communications Letters*, Vol. 21, No. 5, pp. 1063–1066, 2017.
22. Y. Zou, J. Zhu, L. Yang, Y. Liang and Y. Yao, Securing physical-layer communications for cognitive radio networks, *IEEE Communications Magazine*, Vol. 53, No. 9, pp. 48–54, 2015.
23. M. J. Saber and S. M. S. Sadough, Multiband cooperative spectrum sensing for cognitive radio in the presence of malicious users, *IEEE Communications Letters*, Vol. 20, No. 2, pp. 404–407, 2016.
24. Q.-T. Vien, H. Nguyen and A. Nallanathan, Cooperative spectrum sensing with secondary user selection for cognitive radio networks over Nakagami-m fading channels, *IET Communications*, Vol. 10, No. 1, pp. 91–97, 2016.
25. S. Nallagonda, S. D. Roy and S. Kundu, Cooperative spectrum sensing with censoring of improved energy detector based cognitive radios in Rayleigh faded channel, *International Journal of Wireless Information Networks*, Vol. 21, No. 1, pp. 74–88, 2013.
26. L. Khalid and A. Anpalagan, Adaptive assignment of heterogeneous users for group-based cooperative spectrum sensing, *IEEE Transactions on Wireless Communications*, Vol. 15, No. 1, pp. 232–246, 2016.
27. H. V. Poor, *An Introduction to Signal Detection and Estimation*, SpringerBerlin, 2010.
28. Q. Zhao and A. Swami, *Cognitive Radio Communications and Networks: Principles and Practice*. Elsevier, 2009, ch. Spectrum Sensing and Identification.
29. S. M. Kay, *Fundamentals of Statistical Signal Processing, Volume 2: Detection Theory*. Prentice Hall, 1998, ch. Statistical Decision Theory I.
30. K. S. Shanmugan and A. M. Breipohl, *Random Signals: Detection, Estimation and Data Analysis*. Wiley, 1988, ch. Signal Detection.
31. K. Gurney, *An Introduction to Neural Networks*. CRC Press, 1997, ch. Neural Networks - An Overview.
32. K. Du and M. N. S. Swamy, *An Introduction to Neural Networks*. Springer, 2014, ch. Neural Networks and Statistical Learning.
33. MATLAB, "Mlp neural network with backpropagation," [Online]: https://de.mathworks.com/matlabcentral/fileexchange/54076-mlp-neural-network-with-backpropagation.
34. D. Pu and A. Wyglinski, *Digital Communication Systems Engineering with Software-defined Radio*, ser. Artech House mobile communications library. Artech House, 2013. [Online]. Available: http://books.google.com/books?id=7Y-pMQEACAAJ.
35. A. Papoulis and S. U. Pillai, *Probability, Random Variables and Stochastic Processes*, vol. 4th, McGraw-HillNew York, NY, 2002.
36. T. MathWorks, "Binomial Cumulative Distribution Function[Online]," http://www.mathworks.com/help/stats/binocdf.html.
37. Oracle, "Mysql," http://www.mysql.com/.
38. Microsoft, "Microsoft office," http://products.office.com/en-us/access.
39. K. Gurney, *An Introduction to Neural Networks*, Taylor & Francis IncBristol, 1997.
40. M. M. Gupta, N. Homma and L. Jin, *Static and Dynamic Neural Networks: From Fundamentals to Advanced Theory*, vol. 1st, WileyNew York, 2003.
41. R. Ramakrishnan and J. Gehrke, *Database Management Systems*, vol. 3rd, McGraw-Hill Science/Engineering/MathNew York, 2002.
42. MathWorks, "Usrp support package from communications system," www.mathworks.com/hardware-support/usrp.html.



Di Pu was born in Suzhou, China in 1985. She received her B.S. degree (with distinction) from Najing University of Science and Technology (NJUST), Nanjing, China in 2007 and her M.S. and Ph.D. degree from Worcester Polytechnic Institute (WPI), Worcester, MA, USA in 2009 and 2013. Since June 2013, she has been a system modeling applications engineer of Analog Devices at Wilmington, MA, USA. Di Pu is a recipient of the 2007 Institute Fellowship for pursuing Ph.D. studies at WPI in Electrical Engnineering. She is also a winner of the 2013 Sigma Xi Research Award for Doctoral Dissertation at WPI.

Bengi Aygun received her Ph.D. degree from the Department of Electrical and Computer Engineering, Worcester Polytechnic Institute, Worcester, MA in 2016, B.Eng. degree in electronics and telecommunication engineering from Yildiz Technical University, Istanbul, Turkey, in 2009, and the M.S. degree in electrical and electronics engineering from Bahcesehir University, Istanbul, Turkey, in 2012. Her research interests include wireless communication theory, with particular focus on vehicular networks, multiple-input-multiple-output relay networks, dynamic spectrum access and cognitive radio.

Dr. Alexander M. Wyglinski is an Associate Professor of Electrical and Computer Engineering at Worcester Polytechnic Institute (WPI), Worcester, MA, USA and Director of the Wireless Innovation Laboratory (WI Lab). Dr. Wyglinski received his B.Eng. and Ph.D. degrees in 1999 and 2005 from McGill University, and his M.Sc. (Eng.) degree from Queen's University in Kingston in 2000, all in Electrical Engineering. Throughout his academic career, Dr. Wyglinski has published over 35 journal papers, over 75 conference papers, 9 book chapters, and two textbooks. Dr. Wyglinski's current research activities include wireless communications, cognitive radio, software-defined radio, dynamic spectrum access, spectrum measurement and characterization, electromagnetic security, wireless system optimization and adaptation, and cyber physical systems. He is currently being or has been sponsored by organizations such as the Defense Advanced Research Projects Agency (DARPA), the Naval Research Laboratory (NRL), the Office of Naval Research (ONR), the Air Force Research Laboratory (AFRL) - Space Vehicles Directorate, The MathWorks, Toyota InfoTechnology Center U.S.A., and the National Science Foundation. Dr. Wyglinski is a Senior Member of the IEEE, as well as a member of Sigma Xi, Eta Kappa Nu, and the ASEE.