Mitigation of Primary User Emulation Attacks in Cognitive Radio Networks Using Belief Propagation

Sasa Maric(✉) and Sam Reisenfeld

Department of Engineering, Macquarie University, Sydney, NSW 2109, Australia
sasa.maric@students.mq.edu.au, sam.reisenfeld@mq.edu.au

Abstract. In this paper, we introduce a belief propagation based technique to combat the effects of primary user emulation attacks (PUEA) in Cognitive Radio (CR) Networks. Primary user emulation attacks have been identified as the most serious threat to CR security. In a PUEA, a malicious user emulates the characteristics of a primary user and transmits over idle channels. As a result, secondary users that want to use the channels are tricked into believing that they are occupied and avoid transmitting on those channels. This allows the malicious user to use the channels uncontested. To moderate the effects of PUEA, we propose a defence strategy based on belief propagation. In our solution, each secondary user examines the incoming signal and calculates the probability that it was transmitted from a primary user. These probabilities are known as beliefs. The beliefs at secondary users are reconciled to an agreed decision by comparison to a predefined threshold. The decision is made by a secondary user on whether it is believed that received transmission on a channel originated from a legitimate primary user or from a primary user emulation attacker.

Keywords: Cognitive radio networks · Belief propagation · Primary user emulation attacks · Security

1 Introduction

Traditional spectrum allocation methods allocate spectrum over large geographic regions and time spans to primary users (PUs). Primary users are licensed by a government regulatory office, such as the Federal Communications Commission in the United States. Channels in the licensed spectrum bands are allocated exclusively to primary users and are inaccessible to other users [1]. Users, other than primary users who could potentially use these channels, are called secondary users (SUs). It has been shown that the traditional allocation method of fixed channel allocation to primary users is leading to a very low utilisation across the licensed spectrum [2] [3]. Cognitive Radio, a collection of intelligent methods designed to use the radio spectrum in an efficient and dynamic manner, has been proposed as a solution to the frequency spectrum shortage. Cognitive

M. Weichold et al. (Eds.): CROWNCOM 2015, LNICST 156, pp. 463–476, 2015.
DOI: 10.1007/978-3-319-24540-9_38

Radio proposes to increase the efficiency of radio spectrum use by allowing secondary users to use channels when they are unoccupied by primary users. In this way, the average percentage of time for which the channels are actively carrying communication signals is increased. As a result, the total data throughput for the same bandwidth allocation is also increased. This must be achieved, while bounding the interference to a level which causes negligible degradation to the quality of primary user communications[1].

Despite its tremendous potential, Cognitive Radio is yet to be accepted as the solution to the radio spectrum shortage problem. One of the reasons for this is cognitive radio networks are susceptible to a number of types of jamming attacks. The most exploited area in cognitive radio is the spectrum sensing phase, where secondary users scan the frequency spectrum looking for available channels which are unoccupied by primary users. During this phase, if an attacker is able to mimic the signal properties of a primary user, he would be able to trick secondary users into believing that available channels are being used by primary users. This would result in secondary users vacating channels and leaving them available for malicious users to utilise uncontested. This form of attack is called a Primary User Emulation Attack (PUEA).

The remainder of this paper is organized as follows. In section 2, we introduce our system model. In section 3, our defense strategy based on belief propagation is presented. In section 4, we present our simulation result and analysis. Lastly, In section 5, we conclude the paper.

1.1 Related Work

A number of mitigation techniques have been proposed to combat primary user emulation attacks. The most promising of these use localisation of the transmitter. A number of methods exist for localisation of transmitters. These localisation methods can be classified into two categories: distributed localisation and centralised localisation. The first approach uses secondary user cooperation. This type of method involves secondary users trying to solve the localisation problem individually using information from cooperating nodes. The second approach is the central approach. In this approach nodes are scattered around the network and collect snapshots of the transmitted signal. These measurements are sent to a central node that processes the information and makes a decision on whether the suspect is a legitimate user or an attacker.

Locdef [4] is a localisation method that uses both localisation of the transmitter and signal characteristics to determine if the transmitter is a malicious user or not. The Locdef scheme uses sensor nodes scattered around the network to take snapshots of the incoming Received Signal Strength (RSS) at different locations in the network. These measurements are sent to a central location for processing. By identifying peaks in the RSS, a central node is able to determine the location of the transmitted signal. Locdef uses a three stage verification scheme to determine the validity of the incoming signal. The first stage uses the RSS of the signal to determine if it is coming from a primary user location or not. In the second stage the receiver looks at the energy of the received signal.

The reason for this is that secondary users are not able to transmit at high power levels, whereas primary users often are. If a suspect passes the first two stages, the scheme moves on to the last stage where it compares the signal characteristics of the incoming signal with the known characteristics of the idle primary user. If the characteristics of the incoming signal do not match the known signal characteristics of the primary user, the transmitter is deemed to be a malicious user.

Papers [5] and [6] present two primary user emulation attack mitigation schemes based on authentication and encryption. In [6] the author outlines a centralised scheme in which each primary user is given a unique ID number and a random variable (HM) by a centralised base station. Every time a suspect becomes active, the base station goes through a two-step authentication process to insure that the suspect is a valid primary user. Before a primary user can access the network, the user must send their ID number to the BS for authentication. The primary user ID is compared to a pool of identification numbers that correspond to all primary users in the area. If the ID number corresponds to one of the ID numbers in the pool, the scheme moves on to step two of the authentication process. If it does not, the user is treated as a malicious user and is ignored. The second step of the process is called the information displacement step. In this step the HM variable is multiplied by an encryption matrix which returns a value M that is compared to a set of expected values. If the value corresponds to the expected values, the transmitter is authenticated as a primary user. If it does not, the transmitter is treated as a malicious user and is ignored.

In [1] the author presents a technique based on belief propagation. This technique uses cooperation between secondary users to localise a transmitter. Comparing this to the known location of a primary user each secondary user is able to determine with a certain probability whether the transmitter is a primary user. The author denotes this probability as a belief. Secondary users in the network calculate their own local belief and exchange them to their neighbours. Then, each secondary user calculates a final belief using its own beliefs and all the beliefs from its neighbours. This paper modifies the algorithm described in [1] and suggests a useful procedure for determining whether the received signal originates from an attacker or not. Our paper presents substantial improvements to the algorithm described in [1].

2 System Model and Assumptions

In this section, we describe the basic system model that is used throughout this paper. To model the relationship between the transmit signal power and the received signal power, the author in [1] considers both path loss and log normal shadowing of the channel. Using these assumptions, we define an equation for the received signal strength from a primary user k as:

$$P_{r(PU_k)} = P_{t(PU_k)} d_{PU_k}^{-\alpha} h, \tag{1}$$

where, $P_{r(PU_k)}$ represents the received signal power from primary user k, $P_{t(PU_k)}$ represents the transmit power of the primary user k, d_{PU_k} represents the distance

between a secondary user and a primary user k, h is the shadow fading constant defined as $h = e^{ab}$ where $a = \frac{ln10}{10}$, b is defined as a random Gaussian variable with a mean 0 and variance σ^2, and α is a propagation loss exponent. From Eq. (1) we are able to derive a similar equation to define the received signal power from an attacker as:

$$P_{r(attacker)} = P_{t(attacker)} d_{attacker}^{-\alpha} h_{attacker}, \tag{2}$$

where, $P_{r(attacker)}$ represents the received signal power from the attacker, $P_{t(attacker)}$ represents the transmit power of the attacker, $d_{attacker}$ represents the distance between the attacker and a secondary node and $h_{attacker}$ is a shadowing constant similar to the one used in Eq. (1).

3 Detecting PUEA Using Belief Propagation

3.1 Original Belief Propagation Method

Belief propagation provides high accuracy detection of primary user emulation attacks. In belief propagation, each secondary user performs local observations and calculates the probability that an incoming signal belongs to a primary user. To accurately detect the presence of a malicious user, neighbouring nodes must communicate with each other and exchange local observations. Local observations are exchanged in the form of messages. Each secondary user computes a belief about whether the suspect is a primary user or an attacker according to its own local observations and the sum of all incoming messages from all its neighbours. A final belief is calculated using the sum of all beliefs of all SUs. This final belief is compared to a predetermined threshold. If the final belief is above the threshold, the suspect is deemed to be a primary user. If it is below, the suspect is considered to be a malicious user. The belief propagation framework is based on pairwise Markov Random Fields (MRF)[7].

Relative power observations of secondary users represent a pattern of receive powers generated by the location of the transmit station. The exchange of information between secondary users enables recognition of patterns for the purposes of determining whether or not the transmission originates at a known primary user location. In MRF we define Y_i as the local power observation at secondary user i, and X_i as the state of the suspect observed at user i. If X_i=1 the suspect is a primary user, if X_i=0 the suspect is a malicious user. The local function at user i is defined as $\phi_i(X_i, Y_i)$. The local function represents the observations made by a secondary user i about whether the suspect is a primary user or not. The compatibility function $\psi_{ij}(X_i, Y_j)$ is used to model the relationship between secondary users. The higher the compatibility function between two users is the more relevant the local observations of the two users become to each other. For example, if SU_1 is 1m away from SU_2 and SU_1 is 30m away from SU_3, then local observations that come from SU_2 to SU_1 will contribute more to the final belief of SU_1 than local observations that come from SU_3. The joint probability distribution of unknown variable X_i is given by:

$$P(\{X_i\},\{Y_i\}) = \prod_{i=1}^{I} \phi_i(X_i,Y_i) \prod_{i\neq j} \psi_{ij}(X_i,Y_j), \tag{3}$$

where, I denotes the number of SUs in the network. We aim to compute the marginal probability at secondary user i, which we denote as the belief. The belief at a secondary user i is given in Eq. (4). It is the product of the local function at user i and all messages coming into user i from all the neighbours of i:

$$b_i(X_i) = k\phi_i(X_i,Y_i) \prod_{i\neq j} m_{ij}(X_i), \tag{4}$$

where, m_{ij} is a message from a secondary user i to a secondary user j and, k is a normalisation constant that insures that the beliefs sum to 1. Therefore:

$$k = \frac{1}{\prod_{i\neq j} m_{ij}(1)}. \tag{5}$$

In order to compute the belief at each user, we introduce a message exchange equation that is used to iteratively update the belief at each secondary user. In the l_{th} iteration a secondary user i sends a message $m_{ij}^{l}(X_i)$ to secondary user j which is updated by:

$$m_{ij}^{l}(X_i) = C \sum_{X_i} \psi_{ij}(X_i,Y_j)\phi_i(X_i,Y_i) \prod_{k\neq i,j} m_{ki}^{l-1}(X_i), \tag{6}$$

C is another normalisation constant such that $m_{ij}(1)+m_{ij}(0) = 0$, and therefore:

$$C = \frac{1}{\prod_{k\neq ij} m_{ij}^{l-1}(1)(\psi_{ij}(1,0)+\psi_{ij}(1,1))}. \tag{7}$$

Finally, after all secondary users finish computing their beliefs, these beliefs are added up and averaged to derive a final belief. The final belief is then compared to a predefined threshold. If the final belief is higher than the threshold, the suspect is believed to be a primary user. If the final belief is lower than the threshold the suspect is believed to be a malicious user:

$$\begin{aligned} &Honest, && \frac{1}{M}\sum_{i=1}^{M} b_i \geq b_\tau \\ &Malicious, && \frac{1}{M}\sum_{i=1}^{M} b_i < b_\tau, \end{aligned} \tag{8}$$

where, M is the total number of secondary users in the network, $\sum_{i=1}^{M} b_i$ denotes the sum of all the beliefs of all the secondary users on the network and b_τ denotes the pre-set threshold. It is possible that some users would relay false information to other users in the network. However, false information by a small number of nodes would not influence the final belief value significantly.

Local Function. The local function represents the local observations at a single secondary user. Each secondary user calculates its own local function which corresponds to a probability of a suspect being a primary user. To calculate the local function we must compute two probability density functions (PDFs). The first PDF is computed using the RSS measurements that are acquired from the primary user and is denoted by PDF_{pu_k}. The second is a PDF that is computed using RSS measurements acquired from the attacker and is denoted by $PDF_{attacker}$. The local function corresponds to the similarity between the two PDFs. If the PDFs are the same the local function returns a probability equal to 1, which indicates that the suspect is transmitting from a primary user location. The further apart the distributions are the lower the local function and the higher the probability that the suspect is an attacker. The received signal from the primary user can be obtained using the following equation:

$$\frac{P_{r1(PU_k)}}{P_{r2(PU_k)}} = \left(\frac{d_{1(PU_k)}}{d_{2(PU_k)}}\right)^{-\alpha} \left(\frac{h_{1(PU_k)}}{h_{2(PU_k)}}\right), \tag{9}$$

where, $P_{r1(PU_k)}$ and $P_{r2(PU_k)}$ are the RSS values from a primary user(PU_k) to SU_1 and SU_2, $d_{1(PU_k)}$ and $d_{2(PU_k)}$ are the distances between PU_k and SU_1 and SU_2. $h_{1(PU_k)}$ and $h_{2(PU_k)}$ represent the shadow fading between PU_k and secondary users SU_1 and SU_2. It is assumed that the channel response is a circular Gaussian variable $\mathcal{CN}(0,1)$. If we define q as:

$$q = \frac{h_{1(PU_k)}}{h_{2(PU_k)}}, \tag{10}$$

we can then define $B_{i,j}$ as:

$$B_{i,j} = \left(\frac{d_{1(PU_k)}}{d_{2(PU_k)}}\right)^{-\alpha}, \tag{11}$$

therefore, the primary user's PDF of q can be written as follows:

$$PDF_{PU_k}(q) = \frac{1}{\mid B_{i,j} \mid} \frac{2\frac{q}{B_{i,j}}}{((\frac{q}{B_{i,j}})^2 + 1)^2}. \tag{12}$$

The PDF for the attacker is defined in a very similar way to the PDF of a primary user. SUs collect RSS measurements which they then exchanged with their neighbours. We define $P_{r1(attacker)}$ and $P_{r2(attacker)}$ as the received signal strength from the attacker to SU_1 and SU_2 respectively, and the distances between SU_1 an SU_2 and the attacker as $d_{1(attacker)}$) and $d_{2(attacker)}$) respectively. We can then define the value of A as follows:

$$A_{i,j} = \left(\frac{d_{1(attacker)}}{d_{2(attacker)}}\right)^{-\alpha} = \frac{P_{r1(attacker)}/P_{r2(attacker)}}{\pi}, \tag{13}$$

therefore, the attackers PDF can be written as follows:

$$PDF_{attacker}(q) = \frac{1}{\mid A_{i,j} \mid} \frac{2\frac{q}{A_{i,j}}}{((\frac{q}{A_{i,j}})^2 + 1)^2}. \tag{14}$$

To compare the two PDFs we use the Kullback Leibler distance. The Kullback Leibler distance is defined as:

$$KL_{(PDF_{PU_k}, PDF_{attacker})} = \int_0^\infty PDF_{PU_k} log \frac{PDF_{PU_k}}{PDF_{attacker}} dq. \tag{15}$$

The Kullback Leiber (KL) distance calculates the difference between the two PDFs. If the difference between the PDFs is large the KL formula will return a large number and if the distance is small the KL formula will return a small number. To obtain the local function from the KL distance we use the following formula:

$$\phi = \exp(-\min_k KL_{(PDF_{PU_k}, PDF_{attacker})}). \tag{16}$$

The local function returns a probability that a suspect is a primary user. The higher the probability the more likely the suspect is a primary user, the lower the probability the less likely it is that the suspect is a primary user.

Compatibility Function. The compatibility function is essential for cooperation between secondary users. In the belief propagation framework, the compatibility function is a scalar. The higher the compatibility function between two SUs the more relevant the two SUs are to each other. A reasonable compatibility function may be defined by the following expression:

$$\psi_{i,j}(X_i, Y_j) = \exp(-Cd^{\beta}_{X_i, Y_j}), \tag{17}$$

where, C and β are constants and, d_{X_i,Y_j} represents the distance between secondary users i and j. The compatibility function is heavily dependent on the distance between the two secondary users. If the distance between the secondary users is large then the compatibility function tends to zero. If the distance between secondary users is small the compatibility function tents to 1.

The compatibility function is used to insure that SUs that are far away do not have a large contribution to a particular SUs beliefs. The reason for this is that secondary users at different locations suffer from different shadow fading and the more distant users are the less likely to make a significant contribution to the accuracy of a SUs belief. It also insures that closer cooperating SU beliefs have a greater impact on the belief of a SU.

Complete Algorithm. The belief propagation algorithm used in this paper is summarised in Algorithm 1. Each secondary user performs measurements and calculates their PDF_{PU_k} and their $PDF_{attacker}$ using Eq. (11) and Eq. (13). Using these measurements, each secondary user iteratively computes their local

and compatibility functions using Eq. (16) and Eq. (17). Each secondary user then computes and exchanges messages with all its neighbouring nodes. The last step of the algorithm is where each secondary user calculates their belief using their own local observations and the product of all the messages from all their neighbours.

After a number of iterations the mean of all the beliefs is calculated and compared to a predefined threshold. If the final belief is lower than the threshold the suspect is thought of as an attacker, if the final belief is greater than the threshold the suspect is deemed a primary user. The algorithm converges when there is no significant change in the final belief from the previous iteration to the current iteration. Therefore, the algorithm terminates when:

$$\frac{\mid f_b^{l-1} - f_b^l \mid}{f_b^{l-1}} < 0.001, \tag{18}$$

where, $f_b^l = \frac{1}{M} \sum_{i=1}^{M} b_i$, for the lth iteration.This insures that Algorithm 1 converges when there is a change corresponding to less than 0.1% between iterations.

Algorithm 1. Complete defence strategy against the PUEA using belief propagation

1: Each secondary user performs measurements using Eq. (11) and Eq. (13)
2: While $\frac{|f_b^{l-1} - f_b^l|}{f_b^{l-1}} < 0.001$
3: **for** Each iteration **do**
4: Compute the local function using Eq. (16) and the compatibility function using Eq. (17)
5: Compute messages using Eq. (6)
6: Exchange messages with neighbours
7: Compute beliefs using Eq, (4)
8: **end for**
9: Break
10: The PUE attacker is detected according to the mean of all final beliefs based on comparison against threshold.
11: Each SU will be notified about the characteristics of the attacker's signal and ignore them in the future.

3.2 New Belief Propagation Method

This section provides an outline of the changes that were made to the original technique presented in [1]. The two most significant improvements made to the old algorithm are the new simplified local function the new compatibility function.

Local Function. The local function that was used in the original technique suffered from being overly complicated and introducing a high level of complexity into the algorithm making it slow to converge. Our key contribution is the identification of a simpler more efficient local function. The new local function is just as accurate as the previous function. However, instead of doing a large number of numerical evaluations of integrals for each secondary user in the network, the new function calculates a simple arithmetic equation that allows the system to grow linearly instead of exponentially. The new local function that exhibits these desirable characteristics is:

$$\phi_{i,j} = \frac{\mid A_{i,j} - B_{i,j} \mid}{A_{i,j} + B_{i,j}}. \tag{19}$$

The local function is a measure of the similarity between the RSS measurements from a PU and the RSS measurements from a suspect. The closer the correlation between the two RSS values the more likely it is that the suspect is a primary user. The method used to obtain the local function in the old algorithm was computation time intense and had large computational complexity. This was primarily due to the fact that the KL distance was used to calculate the difference between the two probability density functions. The problem with the KL distance is that it uses an integral to determine the dissimilarity between two functions. As the number of secondary users on the network increases, we see a significant difference between the two methods. This is primary due to the fact that the local function has to be evaluated for each pair of secondary users in the network. As the number of SUs in the network increases the number of calculations of the local function increase exponentially. In the sections that follow we present results that prove that our new local function achieves results that are more accurate and efficient than those obtained by the old local function.

Compatibility Function. The compatibility function that was presented in the original paper discouraged cooperation between secondary users in the CR network and as a result decreased the accuracy of the final belief. This was primarily due to the fact that the compatibility function returned values that were very close to zero unless secondary users are located in close proximity. To increase cooperation between SUs we propose the following compatibility function:

$$\psi_{i,j}(X_i, Y_j) = \exp\left(-\frac{d_{X_i,Y_j}}{100}\right). \tag{20}$$

This compatibility function insures that secondary users that are close to each other are able to cooperate and share their result effectively to increase the accuracy of the results. The goal of the modified function is to insure that the messages between secondary users on the network are more relevant. We show in the next section that the new compatibility function is able to improve the performance of the algorithm by allowing a greater degree of cooperation.

4 Simulation Results and Analysis

In this section we present the results of the original BP algorithm against the improved BP algorithm. We chose to use similar simulation parameters as those presented by the authors in [1]. We set the path loss exponent α as 2.5, the transmit power of the secondary user is 0.1W (since the malicious user is also using a cognitive radio this is also the transmit power of the malicious user, we assume this corresponds to a transmission range of about 20 meters). There are 30 secondary users, one primary user and one malicious user deployed in a 100m by 100m grid.

4.1 Original BP Results and Analysis

This section outlines the results that were obtained in [1]. The authors went through a number of scenarios where they moved the locations of the primary and malicious users around the grid. They noted that as the distance between the primary user and malicious user increased, the final belief decreases, meaning that it is easy to distinguish between a primary user and a malicious user if they are far apart. Fig. 1 shows the plot that was obtained using the original BP algorithm.

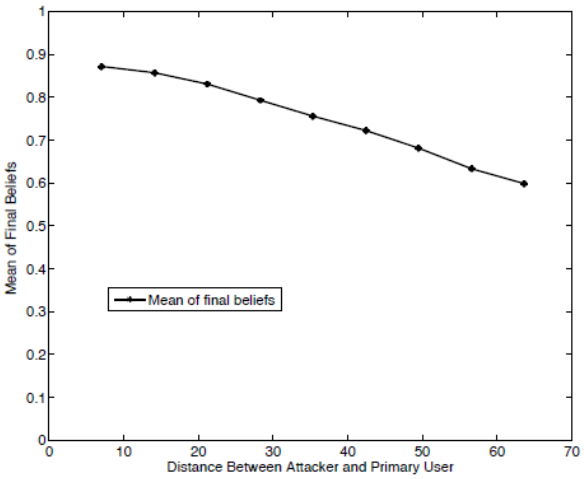

Fig. 1. Final belief Vs Distance (original technique).

We see from the results presented by the authors that the original algorithm is able to distinguish between a legitimate primary user and a malicious user with fairly high accuracy. However, the algorithm that is proposed in the original paper has several deficiencies. The key among these is its high computational

complexity. During our simulations we observed an exponential growth in the computational complexity as the number of secondary users in the network is increased. Fig. 2 shows the effects that increasing the number of secondary users has on the computational complexity.

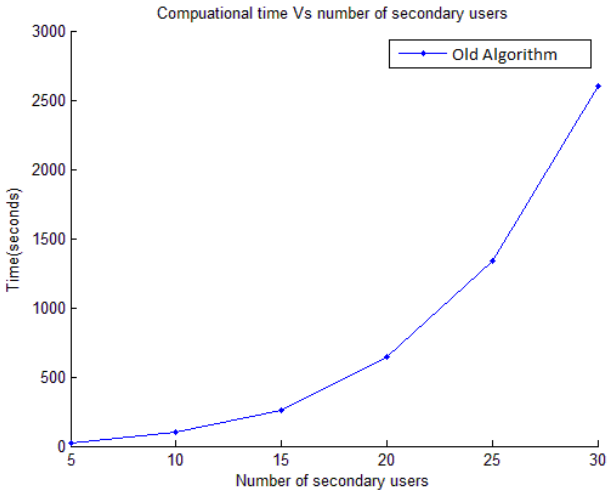

Fig. 2. Computational time of the old technique.

From these results we concluded that although the original algorithm is fairly effective in identifying a malicious user from a primary user, its high computational complexity means that it is not a feasible option for implementation using low power consumption cognitive radio terminals. We identified that the primary reason for the high computational complexity of the original BP algorithm is the computation of the local function. The Kullback Leibler function that is used to evaluate the difference between the primary user probability density function and the attackers probability density function was recognised as the main problem. The reason for this is that the KL function evaluates the difference between two function using an integral expression. If there are n secondary users in the network the KL function has to be evaluated once for each pair of secondary users, which means that it is calculated n^2 times. This is a serious deficiency which makes this algorithm infeasible for practical networks, where the number of users is large.

4.2 New BP Results and Analysis

To combat the deficiencies of the original algorithm, we present a new and improved algorithm that makes two important improvements which increase the accuracy and decease the computational complexity of the original algorithm.

To decrease the computational complexity of the original algorithm we propose a new simplified local function which provides the same level of accuracy with a reduced level of complexity. In addition, we modify the old compatibility function to help increase the level of cooperation between secondary users in the network.

Computational Complexity / Run Time. The most significant improvement obtained by the new technique is the reduced computational complexity and run time of the algorithm. The new algorithm is able to reduce the run time of the original algorithm by a introducing a simplified local function. The new local function insures that the computational complexity grows much slower than in the old algorithm which insures that the algorithm is flexible, scalable and still just as effective. Table 1 presents results that were obtained using an Intel(R) Core(TM) i7-3930k CPU and all simulations were performed and timed using MATLAB.

Table 1. Algorithm run times

Number of users	Comp time Old	Comp Time New
5	22 seconds	0.0491 seconds
10	101 seconds	0.0496 seconds
15	262 seconds	0.0564 seconds
20	648 seconds	0.0682 seconds
25	1337 seconds	0.071 seconds
30	2605 seconds	0.10 seconds

From Table 1 it is clear that the new algorithm is much less computationally complex than the original algorithm. We note that the run times of the new algorithm increase slowly as the number of secondary users in the network is increased. This presents a significant step forward for the algorithm and allows it to be utilised in larger and more complex networks.

Performance and Accuracy. In addition to the reduced computational complexity of the new algorithm, the new algorithm exhibits superior performance to the algorithm presented in [1]. This is primary due to the introduction of a modified compatibility function that allows for a larger degree of cooperation between secondary users. The greater the degree of cooperation between secondary users in the network the lower the chance of false or missed detection of a malicious user. Fig. 3 shows a comparison between the performance of the new algorithm and the performance of the original algorithm.

The perfect BP algorithm would result in a final belief value of 1 when the malicious user and the primary user are at the same location and would result in 0 in all other cases. Through analysis of results we observe that the new algorithm has an average final belief that is smaller than the average of the final belief of the old algorithm. This simple and effective comparison shows that the

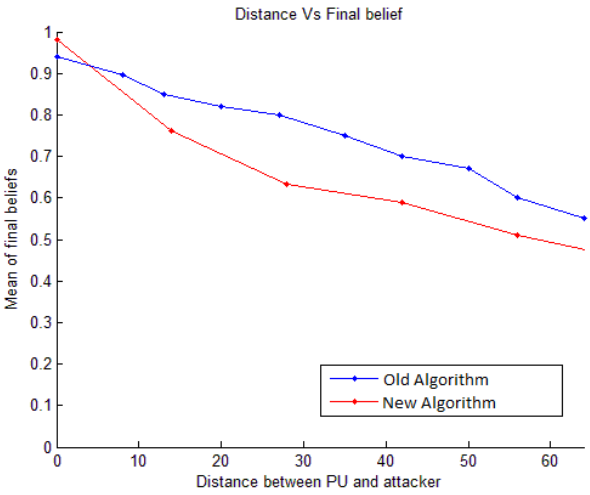

Fig. 3. Comparison of performance between the old and the new techniques.

new algorithm is not just less complicated but also detects PUEA with a higher degree of accuracy.

5 Conclusion

In this paper we present a belief propagation based algorithm to combat the effects of primary user emulation attacks on cognitive radio networks. We introduce key improvements to the algorithm described in [1] in relation to both performance and computational complexity. Through simulation we were able to show that our technique has lower complexity and improved accuracy relative to the technique in [1]. We have shown that the new technique reduces the time of convergence of the BP algorithm from hours to less than a few seconds. Furthermore, despite the simplification of the algorithm we were able to accurately distinguish between primary user and primary user emulation transmissions. These improvements are a direct result of the new local and compatibility functions, which reduce complexity and allow a greater degree of cooperation between secondary users on the CR network. The new algorithm is scalable, efficient, and effective and may be implemented in a low complexity secondary user terminal. The new algorithm provides a significant step forward in the mitigation of primary user emulation attacks in cognitive radio networks using belief propagation.

References

1. Yuan, Z., Niyato, D., Li, H., Song, J.B., Han, Z.: Defeating primary user emulation attacks using belief propagation in cognitive radio networks. Selected Areas in Communications **30**(10), 1850–1860 (2012)

2. Hossain, E., Niyato, D., Han, Z.: Dynamic spectrum access in cognitive radio networks. Chapter 2, 39–72, June 2009
3. Mitola III, J., Maguire Jr., G.Q.: Cognitive radio: Making software radios more personal. IEEE Personal Communications **6**, 13–18 (1999)
4. Chen, R., Park, J.-M., Reed, J.H.: Defense against primary user emulation attacks in cognitive radio networks. Selected Areas in Communications **26**(1), 25–37 (2008)
5. Thanu, M.: Detection of primary user emulation attacks in cognitive radio networks. Collaboration Technologies and Systems **26**(4), 605–608 (2012)
6. Zhou, X., Xiao, Y., Li, Y.: Encryption and displacement based scheme of defense against primary user emulation attack. Wireless, Mobile & Multimedia Networks **4**, 44–49 (2011)
7. Yedidia, J.S., Freeman, W.T., Weiss, Y.: Understanding belief propagation and its generalizations. Exploring Artificial Intelligence in the New Millennium, Chapter 8, 2282–2312, November 2012