# A Context-Aware Trust Framework for Resilient Distributed Cooperative Spectrum Sensing in Dynamic Settings

Aida Vosoughi, *Student Member, IEEE*, Joseph R. Cavallaro, *Fellow, IEEE*, and Alan Marshall, *Senior Member, IEEE*

***Abstract*—Cognitive radios enable dynamic spectrum access where secondary users (SUs) are allowed to operate on the licensed spectrum bands on an opportunistic noninterference basis. Cooperation among the SUs for spectrum sensing is essential for environments with deep shadows. In this paper, we study the adverse effect of insistent spectrum sensing data falsification (ISSDF) attack on iterative distributed cooperative spectrum sensing. We show that the existing trust management schemes are not adequate in mitigating ISSDF attacks in dynamic settings where the primary user (PU) of the band frequently transitions between active and inactive states. We propose a novel context-aware distributed trust framework for cooperative spectrum sensing in mobile cognitive radio ad hoc networks (CRAHN) that effectively alleviates different types of ISSDF attacks (Always-Yes, Always-No, and fabricating) in dynamic scenarios. In the proposed framework, the SU nodes evaluate the trustworthiness of one another based on the two possible contexts in which they make observations from each other: PU absent context and PU present context. We evaluate the proposed context-aware scheme and compare it against the existing context-oblivious trust schemes using theoretical analysis and extensive simulations of realistic scenarios of mobile CRAHNs operating in TV white space. We show that in the presence of a large set of attackers (as high as 60% of the network), the proposed context-aware trust scheme successfully mitigates the attacks and satisfy the false alarm and missed-detection rates of $10^{-2}$ and lower. Moreover, we show that the proposed scheme is scalable in terms of attack severity, SU network density, and the distance of the SU network to the PU transmitter.**

***Index Terms*—Cognitive radio, context awareness, cooperative systems, mobile ad hoc networks, network security, radio spectrum management, wireless networks.**

Manuscript received December 24, 2016; revised March 24, 2017; accepted May 28, 2017. Date of publication; date of current version. This work was supported in part by the U.S. National Science Foundation under Grant ECCS-1408370, Grant CNS-1265332, and Grant ECCS-1232274, and in part by the U.S.–Ireland R&D Partnership USI033 "WiFiLoc8" grant involving Rice University (USA), University College Dublin (Ireland), and Queen's University Belfast (Northern Ireland). The review of this paper was coordinated by Dr. X. Huang. *(Corresponding author: Aida Vosoughi.)*

A. Vosoughi and J. R. Cavallaro are with the Department of Electrical and Computer Engineering, Rice University, Houston, TX 77005 USA (e-mail: vosoughi@rice.edu; cavallar@rice.edu).

A. Marshall is with the Department of Electrical Engineering and Electronics, University of Liverpool, Liverpool L69 3GJ, U.K. (e-mail: alan.marshall@liverpool.ac.uk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVT.2017.2716361

## I. Introduction

The dynamic spectrum access (DSA) paradigm, enabled by cognitive radios, facilitates flexible and efficient spectrum usage by allowing secondary users (SUs) to use licensed spectrum bands of primary users (PUs) on an opportunistic non-interference basis [1]. The SUs must perform spectrum sensing in order to avoid interference with the PUs. Cooperative spectrum sensing (CSS) that exploits the spatial diversity in the SU network effectively relaxes the sensitivity requirements on individual SUs and improves the overall sensing performance [2]. Distributed cooperative spectrum sensing (DCSS) is preferred to a centralized scheme (with a fusion center) as it is scalable, fault-tolerant and more efficient [3]. DCSS also enables cooperative sensing in cognitive radio ad hoc networks (CRAHN) where there is no base station or infrastructure. The existing DCSS schemes which are inspired by distributed average consensus algorithms are based on iterative diffusion and aggregation of data through linear iteration-based or gossip-based schemes and involve communication with direct neighbors in the network graph [4]–[6].

Spectrum Sensing Data Falsification (SSDF) [7] is a known attack for cooperative spectrum sensing schemes, where malicious SUs broadcast falsified sensing data to their neighbors in order to mislead them and compromise the spectrum sharing in the cognitive radio network. SSDF attack can cause the SUs to make incorrect decisions about the PU activity which will result in increased interference from the SUs to the PU and will also lead to underutilization of the free spectrum. Insistent SSDF (ISSDF) attack [8], [9], in particular, is aimed at iterative DCSS schemes where the attacker not only falsifies its sensing data but it also broadcasts the falsified value in every iteration of the cooperation and refrains from updating its value according to the iterative protocol. Thus, ISSDF attacks can be very harmful. Fig. 1 depicts the behavior of three main types of attackers that have been considered for CSS namely fabricating, Always-Yes, and Always-No [10], [11]. Always-Yes and Always-No attackers constantly broadcast high and low power values as their sensing reports, respectively, regardless of the PU activity state. In contrast, a fabricating attacker generates a falsified low or high value indicating the opposite of the true PU activity state.

Distributed trust schemes have been recently introduced for DCSS that require each SU node to maintain a single sliding observation vector per each SU [12], [13]. Whenever an SU node $i$ receives a value from another node $j$, node $i$ compares

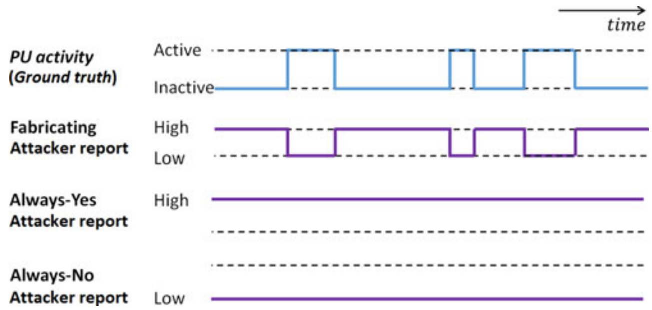



Fig. 1. PU dynamic settings and different types of attackers.

the reported value from $j$ with its own decision about the PU state. Based on this evaluation, node $i$ tags the observation from node $j$ as either an agreement or a conflict and records that in the corresponding observation vector. The trust score that node $i$ assigns to $j$ is then calculated based on the ratio of agreements over the total number of observations (the length of the observation vector) [12], [13]. We call the above trust derivation approach "context-oblivious" as the SU nodes do not distinguish between the observations based on the current PU activity context. Instead, they make blind observations and record all of the observations in a single observation vector regardless of the context.

We will show in this paper that the existing context-oblivious trust schemes are vulnerable to ISSDF attacks in dynamic settings, where the PU of the spectrum band transitions between active and inactive states over time. Thus, these techniques cannot protect the SUs and accordingly the SU nodes make incorrect detection decisions which are harmful to both the primary and secondary users of the spectrum.

Fig. 2(a) shows an example of the vulnerability of the existing agreement/conflict context-oblivious trust schemes. The Always-Yes attacker broadcasts high values (as its sensing report) all the time, even when the PU is active; therefore, in an active cycle (the duration when the PU is active), an honest node will most likely be in agreement with the Always-Yes attacker. Thus, the attacker seems to be non-malicious in the view of the honest node. As a result, the attacker is highly trusted at the end of an active cycle. Fig. 2(b) shows that in an inactive cycle, the Always-Yes attacker who has earned high trust in the previous active cycle is able to deceive the honest node to believe that the PU is active. As a result, the honest SU refrains from using the free channel. This increased false alarm rate among the honest SUs leads to no utilization or underutilization of the free spectrum which is very harmful to the SU network. The context-oblivious trust schemes have a similar vulnerability in mitigating Always-No attackers in dynamic settings, as the trust of Always-No attackers is increased in the PU inactive cycles.

In this paper, we show the vulnerability of the existing trust management schemes in dynamic settings are due to the fact that these schemes are context-oblivious. In order to solve the above-mentioned problem and to mitigate the attacks effectively, we present the following contributions:

1) To the best of our knowledge, this paper is the first to introduce a context-aware trust scheme for DCSS in a mobile CRAHN that is resilient to ISSDF attacks in dynamic settings where the PU frequently transitions between active and inactive states. In our proposed scheme, the trust observations are distinguished based on the speculated context: PU-Present or PU-Absent context. Thus, the trust evaluation of a peer SU is significantly more effective than the current context-oblivious schemes because it is done in a more informed manner.

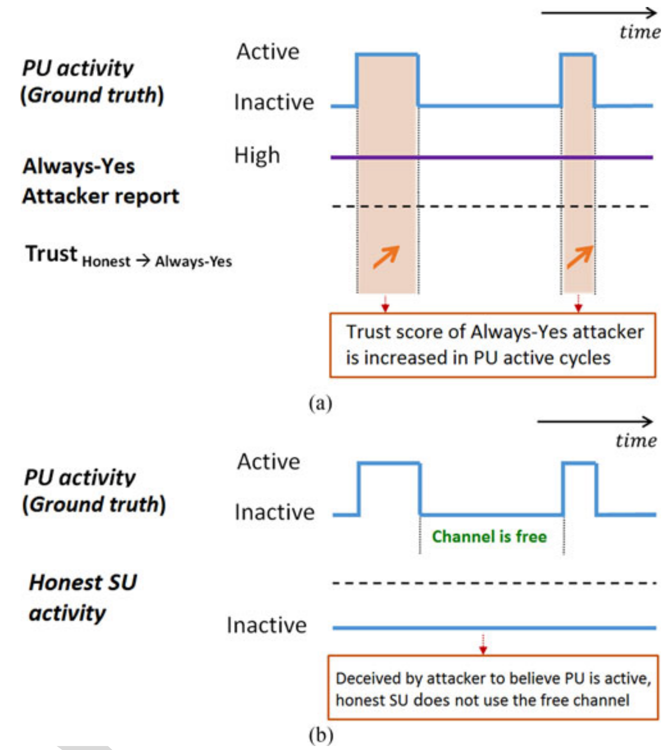



Fig. 2. An Always-Yes attack scenario in PU dynamic settings and vulnerability of the existing (context-oblivious) schemes: (a) The trust score of the Always-Yes attacker is increased when PU is active. (b) In the PU inactive cycle, the highly trusted attacker deceives the honest SU to believe PU is active; thus, the honest SU remains inactive and does not use the free channel.

2) We present a theoretical analysis to evaluate the agreement probability (thus, the level of trust) between the honest nodes and the attackers in the presence of different types of ISSDF attacks (Always-Yes, Always-No, fabricating) and considering the honest mistakes of the honest nodes. The analysis is presented for both the context-oblivious and the proposed context-aware trust schemes.
3) With both theoretical analysis and extensive Monte Carlo simulations, we show that the introduced context-aware trust scheme significantly increases the resilience of iterative DCSS schemes to ISSDF attacks in dynamic settings. Adopting the proposed trust scheme enables a mobile SU network with 20% malicious nodes in a realistic and dynamic environment to satisfy the false alarm and missed-detection rates as low as $10^{-3}$. For a similar scenario, the existing trust schemes cannot even achieve an error rate of $10^{-1}$ regardless of the detection threshold.
4) We show that our proposed trust framework is able to effectively mitigate Always-Yes, Always-No and fabricating attacks in different scenarios with high level of attack severity, even when the majority of the nodes in

the network are malicious. In addition, we show that our proposed scheme is scalable in terms of network density and the distance from the PU transmitter.

## II. Related Work

The conventional SSDF attacks and mitigation approaches against them have been well-studied in the literature for the centralized CSS schemes [7], [10], [11], [14]–[18]. A known mitigation technique against SSDF attacks is that each node assigns history-based trust scores to its neighbors and it weights their sensing reports according to the scores [7]. Recently, average consensus algorithms including gossip-based protocols and linear iteration-based schemes have been used for the DCSS applications [3], [19]–[23]. However, ISSDF attack in the iterative DCSS schemes is hardly explored.

ISSDF attackers are similar to stubborn agents [24], who have fixed opinions and do not update their beliefs based on other agents' opinions. It is shown that the initial opinion of the normal (not stubborn) agents have essentially no impact on the long-run opinion distribution [24]. Sundaram *et al.* [25] also consider a similar attack model aimed at distributed function calculation using linear iterations where the attackers do not follow the iterative update protocol and instead arbitrarily update their values in each iteration. It is shown that the network graph connectivity is a key factor in resilience to these malicious nodes [25]. However, the attack introduced in [25] is different from the ISSDF attack in that the attackers do not change (falsify) their initial values to affect the cooperation.

A trust-aware gossip-based DCSS scheme has been proposed in [20]; however, it does not consider ISSDF attacks and does not benefit from the broadcast nature of wireless and it considers sharing of binary decisions among the nodes. A proposed approach to mitigate the ISSDF attackers in iterative DCSS schemes is outlier detection [26], [27] which is based on detecting the nodes that broadcast values that are deviated from the rest of the neighbors in each iteration. However, this approach requires every node to compute a deviation threshold at each iteration which imposes a significant computational overhead on each SU. In contrast, in our proposed scheme, as will be explained, the SUs update the trust scores only once the consensus iterations are completed and therefore the computational overhead is low. Liu *et al.* [13] propose a trust scheme using trust propagation and a set of pre-trusted nodes to mitigate the effect of Byzantine adversaries in linear iterative consensus in sensor networks. However, trust propagation is costly and generally there are no pre-trusted nodes in an ad hoc network.

A distributed and low-overhead trust management scheme has been proposed recently that is integrated with a consensus-inspired DCSS scheme to mitigate ISSDF attacks [12]. However, this scheme is context-oblivious, and as explained in Section I it cannot mitigate different types of attacks in dynamic settings. To the best of our knowledge, the proposed scheme in this paper is the first context-aware trust scheme for DCSS applications that can effectively mitigate Always-Yes, Always-No and fabricating ISSDF attackers in dynamic settings without the need for centralized or pre-trusted nodes. In addition, our proposed scheme only requires the nodes to perform a single local trust evaluation per sensing round for each direct neighbor, thus the overhead is minimal.

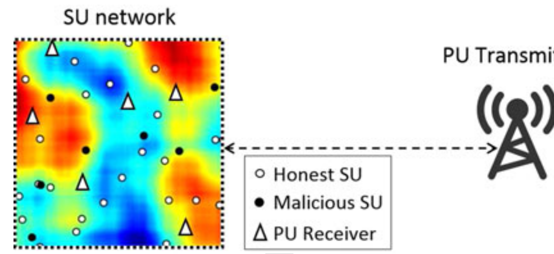



Fig. 3. System overview: Mobile SUs (honest and malicious) are moving in a square location area with diverse shadow fading. Blue represents lower received signal strength from the PU transmitter due to deep shadow fades and red represents higher signal strength.

## III. System Model

We consider a network of $n$ SU nodes that form a mobile CRAHN. The nodes are moving in a square location area within the range of a single stationary PU transmitter which is located outside the square area. Fig. 3 depicts the system overview. Random way point mobility [28] is adopted to model the SU nodes' mobility. A network of PU receivers (either mobile or stationary) may coexist with the SUs in the same location area. Therefore, whenever the PU transmitter is active, the SU's must remain silent to avoid interference to the PU receivers. The detection of a PU transmission is modeled as a binary hypothesis testing problem as follows: $H_0$ if PU is absent and $H_1$ if PU is present. Each SU is equipped with an energy detector to perform spectrum sensing by measuring the received power from the PU transmitter. The received signal by an SU can be modeled as follows:

$$y(m) = \begin{cases} w(m) & H_0 \\ s(m) + w(m) & H_1 \end{cases} \tag{1}$$

where $s(m)$ is the signal component with power $P_S$ and $w(m)$ is the zero-mean additive white Gaussian noise with noise power $P_N$. When the PU is inactive, the sensed power at an SU will essentially be equal to the received noise power. On the other hand, when the PU is active, the signal component power $P_S$ in dB can be modeled as $P_T - PL(d)[dB]$, where $P_T$ is the PU transmission power and $PL(d)$ is the path loss from the PU to the SU located in distance, $d$. If the power detector takes $M$ samples, the test statistic is given by: $\Gamma = \frac{1}{M}\sum_{m=1}^{M} y(m)y(m)^*$. Using the central limit theorem, it can be shown that for large enough $M$ [29], [30], the test statistic for a detector follows a normal distribution [31]:

$$\Gamma \sim \mathcal{N}(P_S + P_N, \frac{2(P_S + P_N)^2}{M}) \tag{2}$$

We model path loss as $PL(d) = \overline{PL}(d) + \psi_{dB}\ \ [dB]$ where $\overline{PL}(d)$ is the average path loss based on the Hata model (suburban areas variant) [32], and $\psi_{dB}$ is a Gaussian random variable in dB with zero mean and a standard deviation of $\sigma_{\psi dB}$ in dB modeling log-normal shadow fading. Therefore the total dB loss is characterized by a Gaussian distribution with mean $\overline{PL}(d)$ and standard deviation $\sigma_{\psi dB}$. The correlation between shadow fading at two locations separated by distance $\delta$ is characterized by $A(\delta) = \sigma^2_{\psi dB} e^{-\delta / X_c}$, where $X_c$ is the decorrelation distance and is usually on the order of the size of the obstacles in the environment [32], [33]. Therefore, closely located receivers (with smaller $\delta$) experience highly correlated shadowing. We model shadows in the environment using random two-dimensional correlated shadow fading maps [34] similar to the example heatmap shown in Fig. 3.

In a non-cooperative scenario, an SU node decides on the PU activity by comparing its own received power test statistic, $\Gamma$, with a detection threshold, $\gamma$. The spectrum sensing performance is characterized by the probability of false alarm ($P_{FA}$) and missed-detection ($P_{MD}$):

$$P_{FA} = Pr(\Gamma > \gamma | H_0) \ \text{ and } \ P_{MD} = Pr(\Gamma < \gamma | H_1) \tag{3}$$

In a distributed cooperative spectrum sensing model, the SU nodes first sense and measure the received power and then share their power measurements with each other to estimate the average received power. After a number of broadcast and update iterations, each SU compares its own estimate of the average power with a threshold to make its final binary decision about the PU presence. We assume a fixed communication range for all of the SU nodes in the network. When a node broadcasts a message, all of the nodes within its predefined radius (one-hop neighbors) will receive that message. Obviously, the neighborhoods are always changing due to the mobility of the nodes; however, we assume that during one sensing period the SU network topology remains unchanged. Here we assume perfect communication between the SUs via a common control channel [35].

In a cooperative spectrum sensing model, a subset of nodes may be malicious. In this paper, we consider the insistent spectrum sensing data falsification (ISSDF) attack model [8]. ISSDF attackers broadcast falsified sensing data to their neighbors in order to cause false alarm or missed-detection errors and to deteriorate the performance of spectrum sensing at the honest (non-malicious) SU nodes. ISSDF attackers do not update their estimates according to the cooperation protocol, instead in order to make the highest impact on the network, they broadcast their falsified values in all of the iterations. We consider three types of ISSDF attackers (Always-Yes, Always-No and fabricating).

In our model, we adopt the trust-aware DCSS scheme introduced in [12]. The iterative update rule is as follows:

$$v_i(c+1) = \theta_{ii}(t) v_i(c) + \frac{\sum_{j \in R_i} \theta_{ij}(t) v_j(c)}{1 + |R_i|}, \ i = 1, ..., n \tag{4}$$

where $v_i(c)$ denotes the value at SU node $i$ at iteration $c$, and $R_i$ is the set of nodes from which node $i$ received a value in this iteration. $\theta_{ij}(t)$ denotes the trust score of node $j$ at the current sensing round $t$ in the viewpoint of node $i$ and the self-trust is $\theta_{ii}(t) = 1 - \frac{\sum_{j \in R_i} \theta_{ij}(t)}{1 + |R_i|}$. The integration of trust scores as weights into the linear iteration-based consensus scheme, makes the combination biased so that the values from more trustworthy neighbors are more effective than the others. The estimation of the trust scores has been the subject of study of many of the previous research works that were mentioned in Section II and different trust schemes have been proposed [7], [10], [12], [13], [18]. In the next section, we introduce our novel distributed context-aware trust framework for trust score derivation which proves to be significantly superior to the previous methods in realistic dynamic settings.

## IV. Proposed Context-Aware Trust Framework

In a realistic cognitive radio network, the primary user of the spectrum band transitions between active and inactive states over time. We show that the dynamics of the PU activity makes the existing context-oblivious trust management schemes (e.g. [12], [13]) vulnerable to ISSDF attacks. In the existing trust schemes, each node records all of its observations from another node in a single observation vector, regardless of the context in which the observations are made.

In contrast, we introduce a context-aware trust management scheme that separates the observations based on the speculated context (PU-Absent or PU-Present). At each sensing round, each SU speculates about the PU activity using all of the available information (from its own sensing and its cooperating neighbors' reports) and conjectures the current context. Based on this speculated context, the SU will record the observations from its neighbors in the corresponding observation vectors. In future sections, we show with analysis and experiments that in realistic dynamic scenarios, our proposed context-aware trust scheme is superior to the existing context-oblivious schemes and can effectively mitigate different types of ISSDF attacks.

Next, we elaborate the proposed context-aware trust scheme. Node $i$ maintains two observation vectors per each peer node $j$: 1) "Absent observation vector", $O^A_{ij}$, 2) "Present observation vector", $O^P_{ij}$. At the end of each sensing round, node $i$ speculates and sets the context based on its own final cooperative decision: either PU-Absent or PU-Present. If at this sensing round node $i$ has received a value from node $j$, node $i$ records the observation from node $j$ based on the context. The observation is recorded in $O^A_{ij}$ if the current context set by $i$ is PU-Absent and in $O^P_{ij}$ if the context is PU-Present. The observation is binary: 0 is recorded if node $i$ and $j$ disagree and 1 is recorded if the two nodes agree on the PU activity in this sensing round. The observation vectors are essentially sliding windows of limited size, thus, if an observation vector is full at the time of recording a new observation, the oldest entry will be discarded. Algorithm 1 describes our proposed context-aware observations for context-aware trust management. $g_{ij}(t)$ denotes the initial value that node $i$ received from neighbor $j$ in the first consensus iteration of sensing round $t$, thus, referring back to (4), $g_{ij}(t)$ is equivalent to $v_j(0)$. The final estimate of node $i$ at sensing round $t$ is denoted by $y_i(t)$ which is equivalent to $v_i(c = \text{final iteration})$ in (4). $\gamma$ denotes the detection threshold.

At sensing round (time) $t$, node $i$ calculates two trust scores, $\theta^A_{ij}(t)$ and $\theta^P_{ij}(t)$ based on the absent and present observation vectors, respectively. Equation (5) shows that the scores are calculated based on the fraction of the observations that are agreements. $H(.)$ denotes the Hamming weight of the binary vector and $|.|$ is the length. The required length of the observation

**Algorithm 1:** Proposed context-aware observation for trust management. Sensing round $t$: Node $i$ observes node $j$:

**if** $(y_i(t) < \gamma)$ **then** `// i sets context: PU-Absent`
  **if** $(g_{ij}(t) < \gamma)$ **then** `// i and j in Agreement`
    $o_{ij}(t) = 1$
  **else** `// i and j in Conflict`
    $o_{ij}(t) = 0$
  **end**
  Add $o_{ij}(t)$ to $O^A_{ij}$; `// Add to Absent Vector`
**else if** $(y_i(t) > \gamma)$ **then** `// i sets context: PU-Present`
  **if** $(g_{ij}(t) > \gamma)$ **then** `// i and j in Agreement`
    $o_{ij}(t) = 1$
  **else** `// i and j in Conflict`
    $o_{ij}(t) = 0$
  **end**
  Add $o_{ij}(t)$ to $O^P_{ij}$; `// Add to Present Vector`
**end**

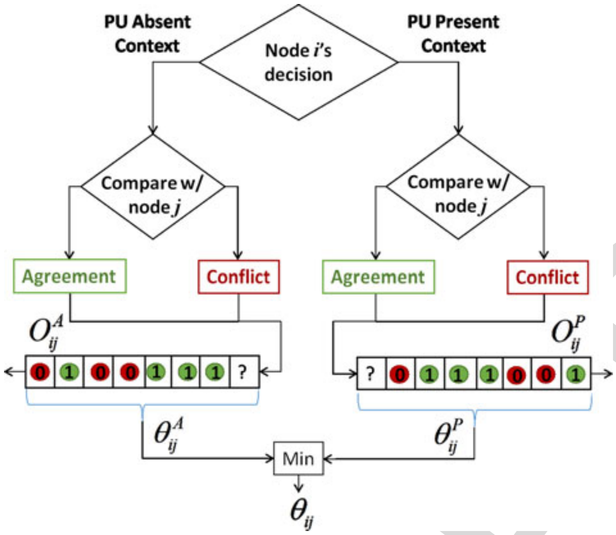



Fig. 4. Proposed context-aware trust scheme: At each sensing round, node $i$ updates the trust score assigned to node $j$ based on the minimum of the scores corresponding to the PU-Absent and PU-Present contexts.

vectors are discussed in Section IV-A in detail. We adopt the zero trust initialization strategy [12] which means the trust scores are initialized to zero and remain zero until the corresponding observation vectors are filled up to the predefined vector length. In addition, the scores are updated only when the final decisions are made at each sensing round and not in between the consensus iterations.

$$\theta^A_{ij}(t) = \frac{H(O^A_{ij})}{|O_{ij^A}|}, \quad \theta^P_{ij}(t) = \frac{H(O^P_{ij})}{|O^P_{ij}|} \tag{5}$$

In each sensing round, node $i$ cannot make its final decision (and set the context) before the cooperation is complete in that round. Therefore, during the cooperation, it cannot know which of the two trust scores ($\theta^A_{ij}(t)$ or $\theta^P_{ij}(t)$) to use for a peer nodes $j$. We propose a conservative approach where the lowest of the two scores is picked as the final trust score:

$$\theta_{ij}(t) = min(\theta^A_{ij}(t), \theta^P_{ij}(t)) \tag{6}$$

Fig. 4 depicts the context-aware trust update algorithm in a flow chart representation showing the procedure of node $i$ updating the trust score assigned to node $j$ at one sensing round. Following the proposed strategy, the honest nodes take no risk and as a result, malicious nodes are always detected and excluded. The conservative score assignment strategy is advantageous because a node that is malicious in one context and not malicious in another context is always assigned a low score corresponding to the context in which it is malicious. As a result, a malicious node will have minimum effect on the honest nodes when it performs its malicious behavior.

Consider the example of an Always-Yes attacker $j$ and let us inspect how it is mitigated by an honest node $i$. Adopting the proposed scheme, all of the observations that $i$ makes from $j$ in the PU-Present context are agreements and all of the observations in the PU-Absent context are conflicts. Therefore, node $i$ perceives that node $j$ seems to be non-malicious in PU-Present context and appears to be malicious in the PU-Absent context. Since all of the observations corresponding to the PU-Absent context are conflicts, the PU-Absent context trust score is zero. Node $i$ assigns the minimum of the PU-Absent and PU-Present scores, which is zero, to $j$. As a result, node $i$ correctly detects the malicious behavior of $j$ and neutralizes its effect. Thus, separating the observations based on the context is necessary to detect the attackers. As we showed before in Fig. 2, for the same example, the context-oblivious schemes are vulnerable and ineffective.

Note that, non-malicious SUs may make honest mistakes and conjecture the context incorrectly due to shadow fading or noise (e.g., see simulation results for non-cooperative scenario in Section VII) which in turn results in an incorrect observation from a peer node. However, the properties of our proposed trust scheme helps the honest SUs to gain trust from one another and to be able to cooperate to correctly conjecture the context at each sensing round. The facilitating properties include evaluation of trust based on averaging over vectors of observations rather than an instantaneous observation and also the zero trust initialization strategy. In addition, since we take a conservative strategy for trust assignment, in case of incorrect context establishment, the malicious SUs cannot gain high trust. We consider these honest mistakes in our theoretical analysis in Section V and in our simulations. Our simulation results presented in Sections VI and VII confirm that the proposed context-aware trust scheme with conservative score assignment is significantly more stable than context-oblivious trust scheme in all of the experimented scenarios.

### A. *Length of the Trust Observation Vector*

Since non-malicious nodes make honest errors, instantaneous observations are not sufficient; thus, as described above, the nodes must make several observations from each other and store them in vectors and rely on the average scores. The honest nodes experience different shadowing and noise levels during time and as they move; therefore, for a sufficiently long observation vector, the average trust scores are more reliable. The shadowing characteristic (decorrelation distance or size of the shadows) and also the mobility characteristics determine the minimum required length of the observation vectors. For example, if the shadows are too large or if the nodes move very slowly, a longer vector may be needed for better trust evaluations between the nodes so that the effect of shadowing can be filtered out. On the other hand, shorter vectors may be preferred in dynamic attack scenarios to achieve fast trust update response to changes in nodes' behavior.

In conclusion, the length of the observation vectors must be determined considering the above trade-offs and the characteristics of the system. For example, as will be described later in Section VI, for our particular simulation setup, we found that the observation vector length of 8 is sufficient. As discussed before, adopting the zero trust initialization strategy, each SU initially does not trust any of the other nodes in the network. An SU can assign a non-zero trust score to another SU as soon as the observation vectors are of length 8 and some of the observations are agreements. However, the trust score of the malicious SUs will remain low because at least in one of the two contexts the conflict rate between the honest node and the attackers is high. The honest nodes then cooperate with their trusted peers to make more accurate final decisions that set the context for the future trust evaluations.

### B. *Mutual Trust Between Two Honest Nodes*

As mentioned before, the honest nodes may make non-malicious mistakes due to fading and noise; therefore, two honest nodes may not agree in their spectrum sensing decisions in a sensing round. In the case of a disagreement between two honest nodes, both of the nodes will decrease the trust score assigned to the other node. Decreasing the score of a non-malicious node that is highly unreliable and reports incorrect data to its neighbors is desired. Such a scenario occurs if there are a subset of honest nodes in the network that experience higher noise or are located in deep shadows and moving very slowly or not moving out of shadow at all. However, in a mobile network, where on average all of the nodes experience the same level of noise and shadowing and have similar mobility characteristics, the average error rates are the same for all of the peer non-malicious nodes.

Therefore, the disagreement between two non-malicious nodes is transient. As discussed before, the trust evaluation based on averaging over a vector of observations filters out these transient mistakes. As a result, over a sufficient number of observations made in both PU-Present and PU-Absent contexts every two normal honest nodes agree with each other more than they disagree. As an example for transient distrust between two honest nodes, consider an honest node $i$ that is located in a shadow area for a while and thus it incorrectly decreases the trust score of an honest neighbor $j$ since they disagree in the PU-Absent context. However, the distrust is transient because as soon as node $i$ moves out of shadow, the two nodes start to agree with each other in the PU-Absent context and $i$ increases the assigned trust score to $j$.

Certainly, there is an inevitable delay associated with the transient effect of the mutual distrust of the honest nodes and this delay will impact the resulting performance negatively. Nevertheless, this is essentially the cost that we pay for trust management to prevent the risk of potential attacks and to mitigate the malicious behavior in the cooperation. As we will show in our analysis and experiments, this negative effect is highly dominated by the positive impact of the trust scheme in detecting and excluding the malicious nodes.

Note that, although we do not explicitly present the mutual trust scores between the honest SUs in the simulation results, in all of our experiments honest nodes do assign trust scores to each other; thus, the presented missed-detection and false alarm rates do include in them the degradation due to the transient distrust. We refer the interested reader to a detailed theoretical analysis and experimental results of the honest-to-honest trust which we have presented in [9, Ch. 6].

## V. Theoretical Analysis of Context-Aware Versus Context-Oblivious Trust

As described in Section IV, a trust score that a node $k_1$ assigns to another node $k_2$ (denoted by $\theta_{k_1,k_2}$) is a measure of the probability of node $k_2$ being honest in the view of $k_1$. Node $k_1$ continuously makes observations from $k_2$ and the trust score is calculated based on the fraction of observations that are agreements. Therefore, the trust score essentially approximates the agreement probability in the most recent set of interactions between the two nodes. In this section, we analyze the agreement probability between the honest nodes and the malicious nodes for both the context-oblivious and the proposed context-aware trust schemes.

### A. *Context-Oblivious Trust Management*

In a context-oblivious trust scheme, node $k_1$ stores its observations from node $k_2$ in a single observation vector $O_{k_1,k_2}$. The event of a node $k_1$ making an observation of node $k_2$ may occur in two conditions: while PU is absent ($H_0$ is true), and while PU is present ($H_1$ is true). Therefore the probability of $k_1$ agreeing with $k_2$ can be written as:

$$\begin{aligned} Pr(agree_{k_1,k_2}) &= Pr(agree_{k_1,k_2}|H_0)Pr(H_0) \\ &\quad + Pr(agree_{k_1,k_2}|H_1)Pr(H_1) \end{aligned} \tag{7}$$

From (7), we can see that if the length of the observation vector is short relative to the PU activity period, then depending on whether $H_0$ or $H_1$ is true, one of the two components in (7) becomes dominant. For example, when PU is absent for a while, all or most of the observations in the observation vector may be from this recent PU inactive cycle and therefore the agreement between the two nodes (and consequently the trust scores) are affected almost only by the probability component corresponding to $H_0$. If the observation vector is much longer than the period of the PU activity, then on average both probability components corresponding to $H_0$ and $H_1$ will have similar effect in the trust score.

In the following paragraphs, we analyze the probability that an honest node $h_1$ agrees with a fabricating, Always-Yes or Always-No attacker. The trust scores that the honest nodes assign to their peers are essentially measured approximations of the agreement probabilities.

*1) Agreement Between Honest and Fabricating:* A fabricating attacker always reports the opposite of the truth about the PU activity. Therefore, when an honest node $h_1$ makes an observation from a fabricating attacker $f_1$, there are two conditions in which the two nodes agree: 1) when $H_0$ is true and $h_1$ makes a false alarm error, 2) if $H_1$ is true and $h_1$ makes a missed-detection error. Equation (8) shows the agreement probability between the two nodes:

$$Pr(agree_{h_1,f_1}) = Pr(F_{h_1})Pr(H_0) + Pr(M_{h_1})Pr(H_1) \quad (8)$$

where, $Pr(F_k)$ and $Pr(M_k)$ of a node $k$ denote the probability of false alarm and missed-detection of node $k$, respectively. If the cooperative decisions of the honest nodes have very low false alarm and missed-detection rates, the agreement rate with the fabricating attacker will be very small as well, thus the assigned trust scores will be small. However, when honest nodes make honest mistakes either in the presence or absence of the PU, in both cases they incorrectly agree with the fabricating attackers and as a result their associated trust scores are increased. For example if PU stays inactive for a while and the honest nodes make many false alarm errors, most of the observations in the observation vector $O_{h_1 f_1}$ are made in $H_0$ and the probability of agreement is essentially close to $Pr(F_{h_1})$ which is high. As a result, when PU finally becomes active, initially, the highly trusted fabricating attackers can significantly affect the detection performance in this cycle.

Therefore, when the context-oblivious strategy is employed, if either missed-detection or false alarm rate of the honest nodes is high, due to deep shadow or high noise, the trust score of fabricating attackers will be increased. We will discuss and show in our simulation results in the next sections that the incorrect increase in trust score of fabricating attackers due to honest mistakes has a destructive effect on the PU detection performance. In contrast, as shown later, the proposed context-aware trust scheme alleviates this problem by considering separate contexts of observations and taking the worst case (the minimum agreement among the two contexts.)

*2) Agreement Between Honest and Always-Yes:* An Always-Yes attacker always broadcasts reports that indicate the presence of the PU. Therefore, an honest node $h_1$ agrees with an Always-Yes attacker, $y_1$, in the following cases: 1) if $H_0$ is true and node $h_1$ makes a false alarm, 2) if $H_1$ is true and $h_1$ does not make a missed-detection error and actually decides that PU is present. Equation (9) derives the agreement probability:

$$Pr(agree_{h_1,y_1}) = Pr(F_{h_1})Pr(H_0) + Pr(\overline{M_{h_1}})Pr(H_1) \quad (9)$$

Obviously, when $H_1$ is true, an Always-Yes attacker's report is indeed correct. Therefore, adopting this context-oblivious trust management scheme, an honest node will incorrectly increase the trust score of an Always-Yes attacker even when the honest node has low error rate (in this case when the honest node does not make missed-detection errors). As we show later, this shortcoming of the context-oblivious trust management is significant and results in the inability of the trust scheme to mitigate Always-Yes attacks.

*3) Agreement Between Honest and Always-No:* Similarly, (10) derives the agreement probability between an honest node, $h_1$, and an Always-No attacker, $n_1$ :

$$Pr(agree_{h_1,n_1}) = Pr(\overline{F_{h_1}})Pr(H_0) + Pr(M_{h_1})Pr(H_1) \quad (10)$$

Therefore, an honest node (with a low false alarm rate) increases the trust score of an Always-No attacker when PU is absent. This makes the context-oblivious trust scheme vulnerable to Always-No attacks.

### B. The Proposed Context-Aware Trust Management Scheme

As described in Section IV, the proposed context-aware trust scheme separates the observations from each node to two contexts, PU-Absent and PU-Present. For both contexts, the event of a node $k_1$ making an observation of another node $k_2$ may occur either when $H_0$ is true or when $H_1$ is true. The context is set by $k_1$'s cooperative final decision which is its best estimate of the PU activity; therefore, "Absent observations" are not necessarily made while $H_0$ is true and "Present observations" are not necessarily made while $H_1$ is true. In this section, we analyze the agreement probability in both PU-Absent and PU-Present contexts to understand the trust scores corresponding to each of these contexts.

When a node $k_1$ makes a cooperative final decision to set the context for its observations, one of the following four events occurs:

1) $B_0^A$: $H_0$ is true and the final decision is PU-Absent. $Pr(B_0^A) = Pr(H_0)Pr(\overline{F_{k_1}})$
2) $B_0^P$: $H_0$ is true and the final decision is PU-Present. $Pr(B_0^P) = Pr(H_0)Pr(F_{k_1})$
3) $B_1^A$: $H_1$ is true and the final decision is PU-Absent. $Pr(B_1^A) = Pr(H_1)Pr(M_{k_1})$
4) $B_1^P$: $H_1$ is true and the final decision is PU-Present. $Pr(B_1^P) = Pr(H_1)Pr(\overline{M_{k_1}})$

Obviously, $B_0^P$ and $B_1^A$ occur when the node makes a false alarm and missed-detection error, respectively. In contrast, in the events $B_0^A$ and $B_1^P$, the node is not in error. We denote the event where the context is set to PU-Absent by $B^A$, which is the union of the events $B_0^A$ and $B_1^A$. Therefore, the probability of $B^A$ can be derived as follows:

$$\begin{aligned} Pr(B^A) &= Pr(B_0^A) + Pr(B_1^A) \\ &= Pr(H_0)Pr(\overline{F_{k_1}}) + Pr(H_1)Pr(M_{k_1}) \end{aligned} \quad (11)$$

Similarly, we denote the event where the context is set to PU-Present by $B^P$, which is the union of the events $B_0^P$ and $B_1^P$. Therefore, we have:

$$\begin{aligned} Pr(B^P) &= Pr(B_0^P) + Pr(B_1^P) \\ &= Pr(H_0)Pr(F_{k_1}) + Pr(H_1)Pr(\overline{M_{k_1}}) \end{aligned} \quad (12)$$

For a node $k_1$, we can derive the following conditional probabilities:

$$Pr(B_0^A|B^A) = \frac{Pr(H_0)Pr(\overline{F_{k_1}})}{Pr(H_0)Pr(\overline{F_{k_1}}) + Pr(H_1)Pr(M_{k_1})} \quad (13)$$

$$Pr(B_1^A|B^A) = \frac{Pr(H_1)Pr(M_{k_1})}{Pr(H_0)Pr(\overline{F_{k_1}}) + Pr(H_1)Pr(M_{k_1})} \quad (14)$$

$$Pr(B_0^P|B^P) = \frac{Pr(H_0)Pr(F_{k_1})}{Pr(H_0)Pr(F_{k_1}) + Pr(H_1)Pr(\overline{M_{k_1}})} \quad (15)$$

$$Pr(B_1^P|B^P) = \frac{Pr(H_1)Pr(\overline{M_{k_1}})}{Pr(H_0)Pr(F_{k_1}) + Pr(H_1)Pr(\overline{M_{k_1}})} \quad (16)$$

We denote the probability of node $k_1$ agreeing with node $k_2$ in the PU-Absent context and PU-Present context by $Pr(agree^A_{k_1,k_2})$ and $Pr(agree^P_{k_1,k_2})$, respectively. These probabilities are written in (17) and (18), respectively.

$$\begin{aligned} Pr(agree^A_{k_1,k_2}) &= \\ Pr(agree_{k_1,k_2}|B^A) &= Pr(agree_{k_1,k_2}|B_0^A)Pr(B_0^A|B^A) \\ &\quad + Pr(agree_{k_1,k_2}|B_1^A)Pr(B_1^A|B^A) \end{aligned} \quad (17)$$

$$\begin{aligned} Pr(agree^P_{k_1,k_2}) &= \\ Pr(agree_{k_1,k_2}|B^P) &= Pr(agree_{k_1,k_2}|B_0^P)Pr(B_0^P|B^P) \\ &\quad + Pr(agree_{k_1,k_2}|B_1^P)Pr(B_1^P|B^P) \end{aligned} \quad (18)$$

*1) Agreement Between Honest and Fabricating:*

*a) PU-absent context:* When honest node $h_1$'s final decision (and thus the context) is PU-Absent, it records its observation from a fabricating node $f_1$ in the "Absent observation vector", $O^A_{h_1,f_1}$. In this context, if $H_0$ is true (the ground truth is that PU is absent), the two nodes definitely disagree since the fabricating node's report indicates that PU is active. On the other hand, if $H_1$ is true, then the two nodes definitely agree, because the fabricating node's report indicates that PU is inactive in this case. Therefore we have the following:

$$\begin{aligned} Pr(agree_{h_1,f_1}|B_0^A) &= 0 \\ Pr(agree_{h_1,f_1}|B_1^A) &= 1 \end{aligned} \quad (19)$$

As a result by replacement in (17), the probability of agreement in the PU-Absent context is equal to the probability that $h_1$ sets the context to PU-Absent while the PU is present which means $h_1$ must make a missed-detection error. Using (14), we have the following:

$$\begin{aligned} Pr(agree^A_{h_1,f_1}) &= Pr(B_1^A|B^A) \\ &= \frac{Pr(H_1)Pr(M_{h_1})}{Pr(H_0)Pr(\overline{F_{h_1}}) + Pr(H_1)Pr(M_{h_1})} \end{aligned} \quad (20)$$

*b) PU-present context:* When honest node $h_1$'s final decision (and thus the context) is PU-Present, it records its observation from a fabricating node $f_1$ in the "Present observation vector", $O^P_{h_1,f_1}$. In this context, if $H_0$ is true, the two nodes definitely agree since the fabricating node reports that PU is active. On the other hand, if $H_1$ is true, the two nodes definitely disagree. Therefore we have the following:

$$\begin{aligned} Pr(agree_{h_1,f_1}|B_0^P) &= 1 \\ Pr(agree_{h_1,f_1}|B_1^P) &= 0 \end{aligned} \quad (21)$$

By replacement in (18), the probability of agreement in the PU-Present context is equal to the probability that $h_1$ sets the context to PU-Present while the PU is absent which means $h_1$ must make a false alarm error. Using (15), we have:

$$\begin{aligned} Pr(agree^P_{h_1,f_1}) &= Pr(B_0^P|B^P) \\ &= \frac{Pr(H_0)Pr(F_{h_1})}{Pr(H_0)Pr(F_{h_1}) + Pr(H_1)Pr(\overline{M_{h_1}})} \end{aligned} \quad (22)$$

The trust score that node $h_1$ assigns to fabricating node $f_1$ is then calculated based on the minimum of the scores derived from the two observation vectors (contexts) as described above (minimum of (20) and (22)).

*2) Agreement Between Honest and Always-Yes:* An Always-Yes attacker, $y_1$, always reports to an honest node $h_1$ that PU is active. Therefore, whenever $h_1$'s final decision is PU-Absent, the observation from $y_1$ is definitely a conflict (probability of agreement is zero) and is recorded in the "Absent observation vector". Note that, the agreement rate is exactly zero regardless of whether $h_1$ sets the context to PU-Absent by mistake (i.e., regardless of the ground truth of the PU activity.) Conversely, whenever $h_1$'s final decision is PU-Present, the observation from $y_1$ is definitely an agreement (probability of agreement is one) and is recorded in the "Present observation vector". As a result, adopting the context-aware trust, an honest node always assigns the minimum trust score which is zero to an Always-Yes attacker and thus can successfully exclude it:

$$\begin{aligned} &Pr(agree^A_{h_1,y_1}) = 0 \\ &Pr(agree^P_{h_1,y_1}) = 1 \\ &min(Pr(agree^A_{h_1,y_1}), Pr(agree^P_{h_1,y_1})) = 0 \end{aligned} \quad (23)$$

*3) Agreement Between Honest and Always-No:* An Always-No attacker, $n_1$, always indicates that PU is inactive to an honest node $h_1$. Therefore, whenever $h_1$'s final decision is PU-Absent, the observation from $y_1$ is definitely an agreement and is recorded in the "Absent observation vector". Whenever $h_1$'s final decision is PU-Present, the observation from $y_1$ is definitely a conflict and is recorded in the "Present observation vector". Taking the minimum, an honest node always assigns a zero trust score to an Always-No attacker, which means the honest node can successfully exclude the attacker:

$$\begin{aligned} &Pr(agree^A_{h_1,n_1}) = 1 \\ &Pr(agree^P_{h_1,n_1}) = 0 \\ &min(Pr(agree^A_{h_1,n_1}), Pr(agree^P_{h_1,n_1})) = 0 \end{aligned} \quad (24)$$

In the next section, we evaluate the probability of agreement between an honest node and an attacker (different types) with simulations in realistic settings. We will show that in all of the simulation scenarios under different types of attacks, adopting the context-aware trust scheme significantly reduces the effect of the attackers by assigning the lowest trust scores to them.

TABLE I
SIMULATION PARAMETERS FOR EVALUATING TRUST-AWARE DCSS

| Path Loss and Shadow Fading | | Random Way Point Model | |
|---|---|---|---|
| PU Dist. from CRAHN | 15 km | CRAHN Area | 200 m × 200 m |
| PU Antenna Height | 30 m | Min Velocity | 1 m/s |
| SU Antenna Height | 1 m | Max Velocity | 2 m/s |
| Center Freq. | 615 MHz | Min Pause | 60 s |
| Log-normal Shadowing SD ($\sigma_{\psi dB}$) | 8 dB | Max Pause | 120 s |
| Decorrelation Dist. ($X_c$) | 50 m | | |
| Transmit Power ($P_T$) | 54 dBm | | |
| **Noise and Threshold** | | **Monte Carlo Simulation** | |
| Noise Figure | 11 dB | # SU Nodes | 25 |
| Channel Bandwidth | 6 MHz | # Consensus Iter. | 4 |
| Noise Power ($P_N$) | −95.22 dBm | SU Node Range | 80 m |
| Threshold ($\gamma$) | [−96, −80] dBm | Simulation Time | 8000 s |
| | | Sense Interval | 2 s |
| | | PU activity period | 800 s |

## VI. SIMULATION RESULTS

In this section, we present the results of our Monte Carlo simulations to evaluate the performance of our proposed context-aware trust scheme in mitigating the effect of different types of attackers. Table I describes our simulation setup. We consider a network of 25 SU nodes that are mobile in a 200 m × 200 m square location area. Each SU node can communicate with any of the other SU nodes located within its 80 m radius. We make the assumption that the sensing frequency of the SUs in the network is much faster than the PU activity frequency: Each SU senses the spectrum every 2 seconds (as recommended in IEEE 802.22 [36]) and the PU's period of activity is 800 seconds with a 50% duty cycle, which means the PU is active for 400 seconds and inactive for 400 seconds periodically. Each Monte Carlo simulation employs a different and randomly generated shadow fading map and it spans 8000 seconds during which the SUs are mobile. In each sensing round, the number of consensus iterations is 4 (See (4): the iterative update.) From the simulation parameters, it can be derived that at any point of time each of the 25 SUs in the network has 11 neighbors on average (for uniformly distributed nodes in the square location area.) Since the nodes are moving in the area, their neighborhoods are constantly changing. The presented results in this section in terms of false alarm and missed-detection performance are averaged over 10000 Monte Carlo runs to ensure sufficient randomness is captured.

As explained in Section IV-A, the minimum required length for the observation vector is determined by the characteristics of the system. According to our experimental results, the length of 8 is sufficient for our system setup and thus we have fixed $O_{\min} = 8$ in our experiments that are presented in this section (no considerable performance improvement was observed using larger observation vector lengths 16 and 32). The length of the observation vector, 8, is small compared to the period of the PU activity. In addition, for this fixed observation vector length, we experimented with smaller PU activity period of 80 s and 8 s and no noticeable difference has been observed in the performance of our proposed context-aware trust scheme. Zero trust initialization is used in all of the experiments unless otherwise stated.

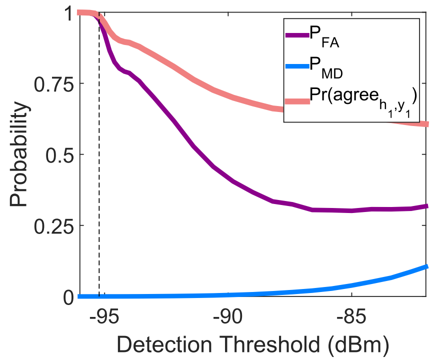
### A. Mitigating Always-Yes Attack

Fig. 5 presents the average false alarm and missed-detection rates from Monte Carlo simulations in a scenario where 20% of the SUs are Always-Yes attackers. The figure also depicts the agreement probability between an honest SU and an Always-Yes attacker based on the analysis in Sections V-A and V-B and using average false alarm and missed-detection rates that are measured from the simulations.

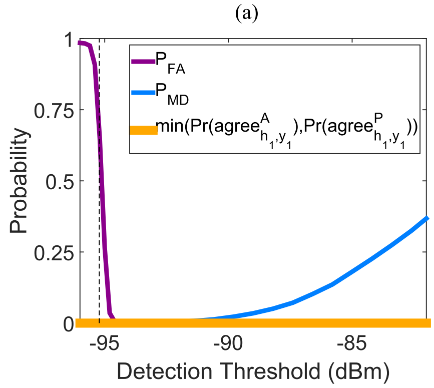
Fig. 5(a) and (b) show the results corresponding to the context-oblivious and the proposed context-aware trust schemes, respectively. The context-oblivious scheme incorrectly assigns high trust scores to the Always-Yes attackers, since in the PU active cycles, the honest SUs agree with the Always-Yes nodes. In addition, for low detection thresholds, where the false alarm rate of the honest SUs is high, they agree with the Always-Yes attackers even in the PU inactive cycles. The agreement with Always-Yes attackers decreases as the threshold increases and false alarm rate decreases.

On the other hand, as discussed in Section V-B2, with the proposed context-aware scheme, an honest node is able to correctly assign the trust score of 0 to an Always-Yes attacker because it takes the minimum of the trust scores in the PU-Present and PU-Absent contexts (See (23).) In Fig. 5(b) only the minimum of the two agreement probabilities is shown which is 0. Thus, as seen from the figure, the proposed scheme effectively mitigates the attack and the false alarm rate sharply drops for the detection thresholds above the average noise power (vertical black dashed line at -95.22 dBm.) Thus, in terms of false alarm error rate, the context-aware trust strategy performs significantly better than the context-oblivious trust strategy.

In terms of missed-detection, the error rate intuitively increases for higher detection thresholds in both the context-oblivious and context-aware schemes. Since the Always-Yes attackers broadcast high values regardless of the PU activity, the malicious behavior of these attackers is advantageous when the PU is present. The reason is that the nodes in shadows might be corrected by cooperating with the Always-Yes nodes. We call this a positive side-effect of the Always-Yes malicious behavior. As a result, excluding the attackers has the counter-intuitive result of higher missed-detection errors. Since the context-oblivious trust strategy is not as effective as the context-aware scheme in mitigating Always-Yes attackers, it results in better missed-detection rate as shown in Fig. 5. Nevertheless, the negative effect of the Always-Yes attackers is significant when PU is inactive and therefore these attackers must be mitigated using the trust scheme. Receiver Operating Characteristic (ROC) curves enable us to fairly evaluate our proposed context-aware trust scheme as we need both of the missed-detection and false alarm error rates to be as small as possible at the same time for a given detection threshold. ROC curves in Fig. 6 show missed-detection and false alarm error rates for a range of detection thresholds (as described in Table I) in two scenarios: 1) in the presence of 20% Always-Yes



(a)



(b)

Fig. 5. 20% Always-Yes ISSDF attack (Vertical dashed lines: Noise power). (a) Context-oblivious trust management. (b) Proposed context-aware trust management.

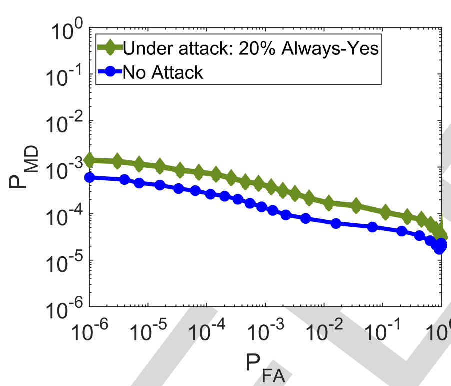
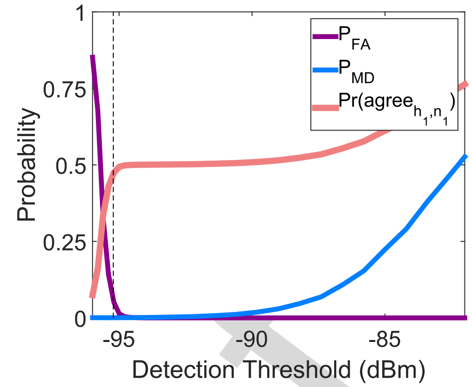
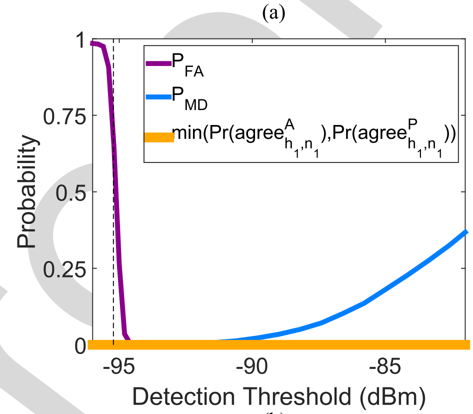



Fig. 6. ROC performance analysis: Resilient DCSS with proposed context-aware trust scheme mitigating Always-Yes ISSDF attack.



(a)



(b)

Fig. 7. 20% Always-No ISSDF attack (Vertical dashed lines: Noise power). (a) Context-oblivious trust management. (b) Proposed context-aware trust management.

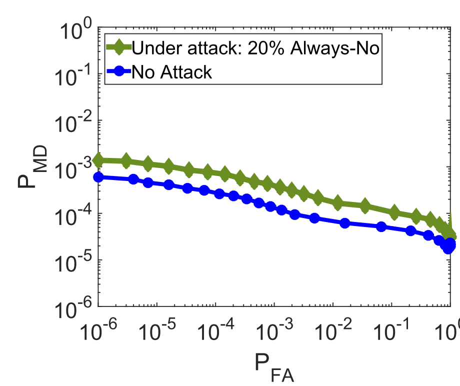



Fig. 8. ROC performance analysis: Resilient DCSS with proposed context-aware trust scheme mitigating Always-No ISSDF attack.

attackers, and 2) with no attackers. It is clear from the ROC plots that the proposed context-aware trust strategy is able to effectively contain the attack and maintain the error rate close to the no-attack case. In conclusion, as the presented ROC curve reveals, our proposed scheme offers sufficiently low missed-detection and false alarm rates at the same time in the presence of Always-Yes attackers.

## B. Mitigating Always-No Attack

Fig. 7 shows the resulting average false alarm and missed-detection rates of the Monte Carlo simulations for the scenario where 20% of the SUs conduct Always-No attacks. The figure presents the results for both context-oblivious and our proposed context-aware trust schemes. It also depicts the agreement probability based on the analysis in Sections V-A and V-B and using average false alarm and missed-detection rates from the simulations.

As can be seen from the figure, in terms of missed-detection error rate, the context-aware trust strategy performs better than the context-oblivious trust strategy. As explained in Section V-A and shown in Fig. 7(a) the context-oblivious trust incorrectly assigns high trust scores to the Always-No attackers as the agreement probability is high in all of the PU inactive cycles. For very low thresholds, where false alarm is too high, the trust score is low but as the threshold increases and the false alarm rate drops, in a PU inactive cycle, the Always-No attackers are in agreement with the honest nodes and as the duty cycle of PU is 0.5 their trust score approaches to 0.5. When the threshold is too high and the missed-detection rate starts to increase, the trust of the Always-No attackers increases even more because now the agreement between the honest nodes and the attackers also occurs in the PU active cycles. On the other hand, as seen from Fig. 7(b), with our proposed context-aware trust management, the honest nodes assign trust of zero to the Always-No attackers (See (24)) and therefore can effectively mitigate them. Similar to the case of the Always-Yes attack, here Always-No attackers have a positive side effect on the false alarm rate, meaning that since they broadcast low values even when PU is absent, they will reduce the chance of false alarms in the network. Therefore, in terms of false alarm rate, at very low thresholds (below the noise power) the context-oblivious scheme performs better than the context-aware scheme . Fig. 8 presents the resulting ROC curve for the scenario 20% Always-No attack. It is clear that the proposed context-aware trust management effectively mitigates the attackers and maintains a performance close to the no-attack scenario.

### C. *Mitigating Fabricating Attack*

Fig. 9 compares the performance results and the agreement probabilities of the context-oblivious and context-aware trust schemes in a scenario where 20% of the SUs are fabricating attackers. As seen from the results, the proposed context-aware trust scheme is superior to the context-oblivious trust in terms of both false alarm and missed-detection.

Fabricating attackers always broadcast a fabricated value that is the opposite of the true sensing measurement. Therefore, if an honest node does not make false alarm or missed-detection mistakes, then in both of the PU active and inactive cycles, the node will be in conflict with a fabricating attacker. However, if the honest nodes do make erroneous final decisions, then adopting the context-oblivious trust scheme, they incorrectly increase the trust of the fabricating attackers (See (8) in Section V-A.) The honest/fabricating agreement in the context-oblivious scheme shown in Fig. 9(a) confirms that for both high false alarm rate (for low thresholds on the left) and high missed-detection rate (for high thresholds on the right), the honest/fabricating agreement is increased. As a result the context-oblivious trust scheme cannot mitigate the impact of the fabricating attackers in these cases. In high false alarm case (due to high noise), the trusted fabricating attackers can increase missed-detection rate and in high missed-detection case (due to deep shadow), the trusted attackers can increase false alarm rate.

On the other hand, the proposed context-aware trust scheme, as shown in Fig. 9(b), picks the minimum of the trust scores associated with "Absent observations" and "Present observations" to filter out the mistakenly high honest/fabricating agreements at the two extremes of the threshold range. As a result, a small trust score, close to zero is assigned to the fabricating attacker. The honest nodes may be unreliable either because they are likely to make missed-detection errors (high detection thresholds relative to the signal strength) or false alarm errors (low thresholds relative to the noise level) but normally not both at the same time. Therefore, by adopting the context-aware trust strategy, the honest nodes will be able to detect the malicious behavior and to update the score of the fabricating attackers correctly. Our proposed context-aware trust management scheme is more cautious, by separating the observations in PU-Present and PU-Absent contexts and picking the minimum of the two scores (See (20) and (22).) The ROC curves in Fig. 10 clearly

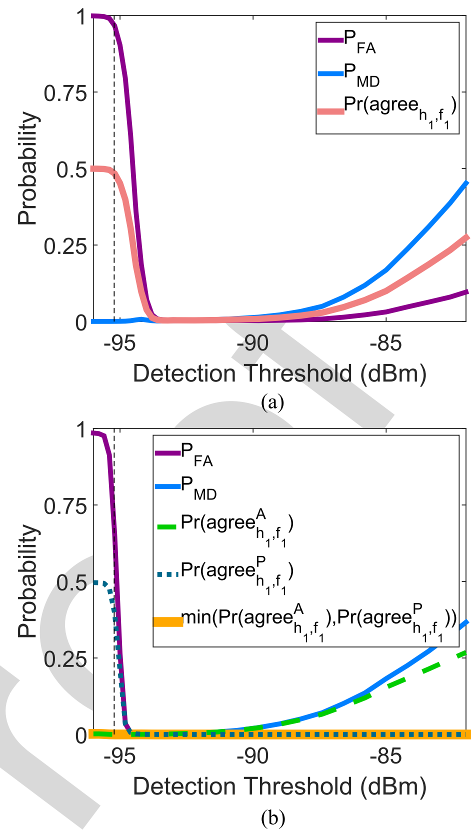



Fig. 9. 20% Fabricating ISSDF attack (Vertical dashed lines: Noise power). (a) Context-oblivious trust management. (b) Proposed context-aware trust management.

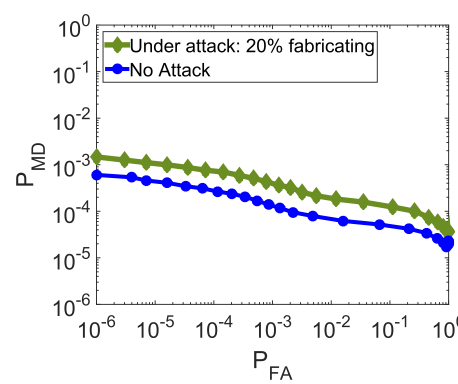



Fig. 10. ROC performance analysis: Resilient DCSS with proposed context-aware trust scheme mitigating fabricating ISSDF attack.

show that the context-aware trust is essential and effective in mitigating the attack in the case of fabricating attack as well.

### D. Discussion on the Simulation Results

The results presented in this section for various scenarios reveal that by adopting the proposed context-aware scheme, the resultant performance is consistent across all of the three types of attacks. As shown in Figs. 5, 7, and 9, unlike the context-oblivious scheme, the context-aware scheme results in the same missed-detection and false alarm rates in all of the three cases by maintaining a trust score of zero or close to zero for the attackers. Similarly, the ROC plots in Figs. 6, 8, and 10, confirm that our scheme offers essentially the same performance for all of the attack cases by successfully neutralizing the attackers (which form 20% of the network). Therefore, the proposed trust scheme offers a comprehensive solution for mitigating different attack scenarios.

In the next section, we continue our analysis and comparison with respect to different characteristics of the network including the attack severity, the SU network density and the distance of the network to the PU. In addition, we analyze the dynamic range of the detection threshold in different scenarios to satisfy a desired performance in the presence of attackers.

## VII. Comparative Performance Analysis

### A. Mitigating Attacks of Different Severity Levels

In Fig. 11 we analyze a few examples of simulation runs that show the progress over time of the average of the trust scores that the honest nodes in the network assign to one typical Always-Yes attacker. For this particular simulation, we have fixed the detection threshold to -93 dBm, a middle threshold where the average error rates for the honest nodes is not at high rates (at this threshold, the measured average probabilities of false alarm and missed-detection for an individual honest node are 0.0007 and 0.0276, respectively.)

The simulation spans 4000 sensing rounds (8000 s) and during this time the nodes are mobile. The shown plots have one data point per 50 s. For this set of experiments, we enforced an initial trust score of 0.5 for all of the nodes (rather than initializing to zero) in order to show how the honest nodes are able to reduce the trust of an attacker from 0.5 to zero and to maintain the zero trust. Fig. 11(a) shows the results where 20% of the nodes are attackers. As expected, in the context-oblivious trust scheme, the trust of the Always-Yes attacker is increased whenever PU is active. The randomness of the trust score is due to the mobility of the nodes and the changes in the neighborhoods; nevertheless, the increase in the trust score in active cycles (shaded areas) is clearly seen. As mentioned before, this is the reason why the context-oblivious trust is not effective in mitigating the attackers.

Note that, with the context-oblivious scheme, the assigned trust to the Always-Yes attacker remains high in most of the inactive cycles (white areas) showing only a small decrease. This clearly shows that once the ISSDF Always-Yes attackers in the network gained increased trust (in the active cycles), they strongly affect the final decisions of the honest nodes in the inactive cycles. Since the honest nodes mistakenly decide PU is active, the Always-Yes attackers appear to be in agreement with the honest nodes which in turn makes the honest nodes believe the attackers are trustworthy. As a result, the trust associated with the attacker is hardly decreased. In contrast, the proposed context-aware trust scheme, successfully reduces the trust of the Always-Yes attacker from the initial trust score down to 0 and keeps it low and therefore effectively excludes

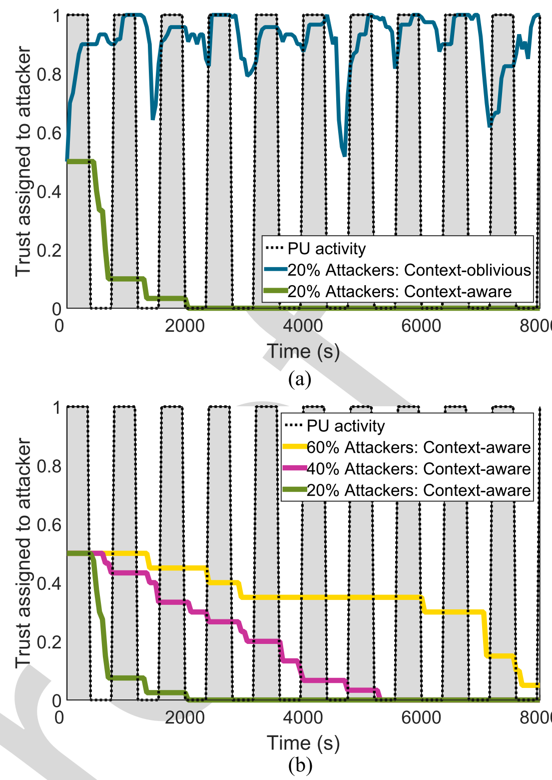



Fig. 11. Average trust score of the honest nodes assigned to a typical Always-Yes attacker. Trust scores initialized to 0.5. Detection threshold = −93 dBm. (a) 20% Always-Yes ISSDF attackers. (b) Different severities of Always-Yes ISSDF attack.

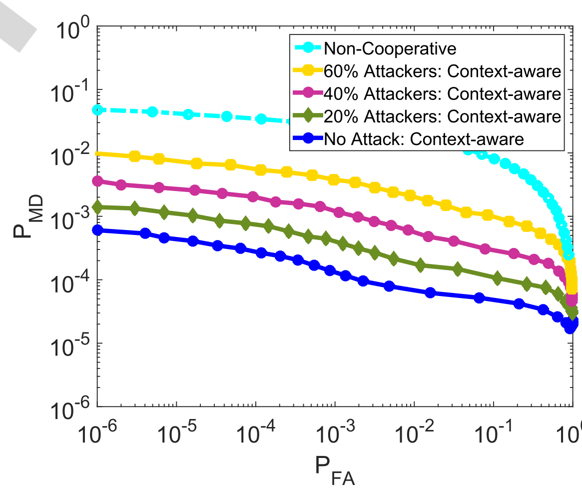



Fig. 12. ROC performance analysis: The proposed resilient DCSS scheme with context-aware trust under various Always-Yes ISSDF attack severity.

the malicious node. Fig. 11(b) compares the trust progress in different attack severity scenarios. All of the attackers in the network are of the same type (i.e., Always-Yes) and thus they strengthen each other's effect. As seen in the plot, the proposed context-aware trust successfully reduces the trust score of the attacker to zero even when the majority of the nodes are attackers (60%). The Always-Yes attackers initially have a trust score of 0.5 in the viewpoint of all of the honest nodes in the network. As the honest nodes observe these attackers, they fill up their observation vectors corresponding to both the PU-Absent and PU-Present contexts. As soon as the number of observations in a vector reaches the predefined minimum (8 observations), the trust score that is calculated based on these observations (5) replaces the initial 0.5 score.

As described in the previous section, whenever the final decision of an honest node is PU-Absent, its observation from an Always-Yes attacker will be a conflict. Therefore, as soon as 8 PU-Absent observations are made from an Always-Yes attacker, the score corresponding to PU-Absent context will be zero and thus the honest node assigns the smaller trust score of the two contexts (which is zero) to the attacker (See (23).) Although in our experiment, the PU is absent half of the time, initially due to the effect of the Always-Yes attackers (with initial trust scores of 0.5), the honest nodes are misled to decide that the PU is present most of the time. As a result, the PU-Absent observation vectors of an honest node get filled-up (i.e. reaches 8 observations) in a longer period of time compared with a no-attack scenario. The more severe the attack is, the attackers are initially more effective and it takes the honest nodes a longer time and a larger number of observations to fill their PU-Absent vectors. As a result, for more severe attacks, the convergence of the trust score towards zero takes a longer time. However, as seen in Fig. 11(b) in all of the attack scenarios including the most severe ones, eventually, the trust is reduced to zero. Therefore, the attackers are completely neutralized.

Fig. 12 shows the ROC results of Monte Carlo simulations of Always-Yes attack scenarios of different severity levels. This figure is an extension to the previously shown Fig. 6, where only 20% attack was considered. The simulation setup is the same as the setup described in Section VI (thus, adopting the zero trust initialization strategy.) The proposed scheme successfully mitigates the attacks in all of the scenarios including the case where the majority of the SUs are malicious. For comparison, we also show ROC of the non-cooperative case where the SU nodes make decisions independently without cooperation. Thus, in the non-cooperative scenario, the SU nodes are greatly affected by shadow fading and noise but they are not affected by ISSDF attackers. By utilizing the context-aware trust scheme, even under the most severe attack (i.e. 60% of the network), the resulting performance of the cooperative spectrum sensing is significantly better than the non-cooperative scenario. Therefore, the trust scheme successfully restricts the destructive effect of the attackers on the cooperation.

As shown before, the performance of the proposed trust scheme is consistent across different types of attacks. Similar results for attack severity scalability are achieved for Always-No and fabricating attacks. These results show that our proposed trust scheme is able to alleviate various attacks of different severity levels, thus, it provides an effective defense system against ISSDF for a wide variety of realistic scenarios.

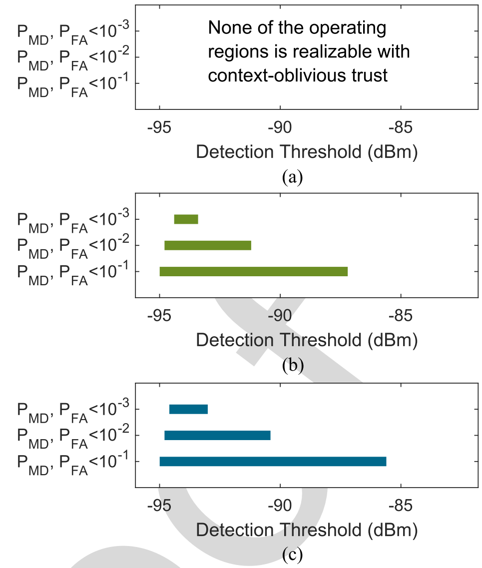



Fig. 13. Range of detection thresholds to realize desired operating regions in terms of $P_{MD}$ and $P_{FA}$. (a) Under 20% Always-Yes attack: With context-oblivious trust. (b) Under 20% Always-Yes attack: With context-aware trust. (c) No attack.

### *B. Enhanced Detection Threshold Dynamic Range*

Fig. 13 compares the ranges of the detection thresholds that satisfy different operating regions in terms of missed-detection and false alarm rates for different scenarios. Under 20% Always-Yes attack, the context-aware trust helps to maintain the dynamic range of the detection threshold to approach to the honest case. In contrast, using the context-oblivious trust scheme, the attack affects the network significantly; as a result, regardless of the detection threshold, none of the operating regions, not even the most relaxed one ($10^{-1}$ error rate) can be achieved. The presented results confirm the significance of the proposed trust scheme in enhancing the flexibility and relaxing the sensitivity requirements of the cognitive radio devices.

### *C. Scalability of the Proposed Trust Scheme*

In this section, we analyze the scalability of the proposed context-aware trust scheme for DCSS in terms of SU network density and distance of the SU network from the PU transmitter. Fig. 14(a) shows the performance results for variable network density for a fixed detection threshold of $-94$ dBm where 20% of the nodes are fabricating attackers. In all of our experiments in the previous sections, we considered 25 SU nodes in a 200 m $\times$ 200 m location area (i.e., density of 625 SUs per km$^2$). In Fig. 14(a), however, the number of SU nodes is varied from only 5 nodes up to 50 nodes in the same area size which results in a density of 125 up to 1250 SUs per $km^2$. Therefore, we consider a variety of scenarios from a sparse to a dense SU network. In this set of simulations we use the same setup (e.g. PU

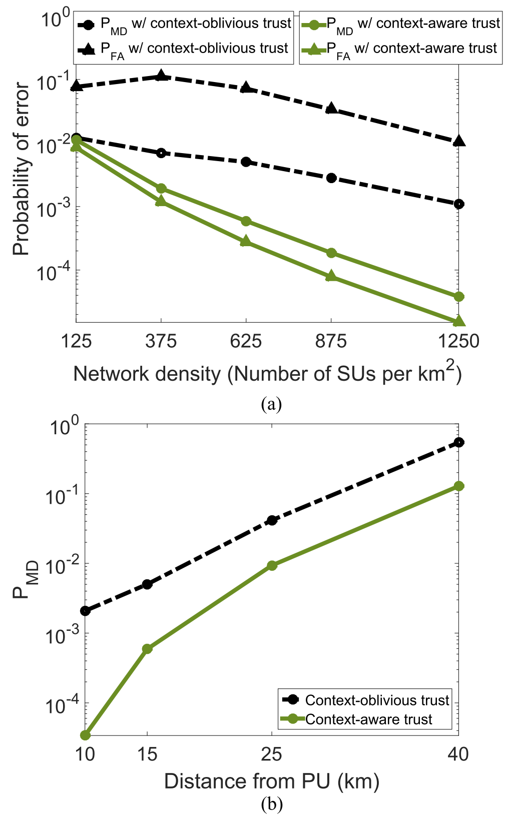



Fig. 14. Scalability analysis in terms of (a) SU network density and (b) distance from PU. Attack scenario: 20% fabricating attackers. Detection threshold = −94 dBm.

activity, SU mobility, 15 km distance from the PU transmitter) as described in Table I.

For a higher SU network density, there are more nodes in the neighborhoods and in general there is more diversity in the network that can be exploited by cooperation. As a result, both $P_{MD}$ and $P_{FA}$ should improve when the density of the network is increased. However, at the same time, in a denser network, the attackers get greater opportunity to propagate their falsified values in the network if they are not properly contained by the trust management scheme. The results presented in Fig. 14(a) shows that using the context-oblivious trust, the false alarm rate of the 375 SUs/km$^2$ case is higher than that of the 125 SUs/km$^2$ case. For denser networks, then the error rates decrease, but both false alarm and missed-detection rates remain relatively high even in the densest case. This confirms that the attackers are not mitigated adequately by the context-oblivious scheme.

In contrast, our proposed context-aware trust scheme limits the impact of the attackers and therefore, increasing the density of the nodes is beneficial as the diversity is increased. In conclusion, our proposed trust-aware DCSS scheme scales well with the network density and performs notably better than the context-oblivious trust regardless of the network density. In fact, as it is clear from Fig. 14(a), the gap between the proposed scheme and the contexts-oblivious scheme becomes more significant for denser networks.

In Fig. 14(b), we analyze the scalability in terms of the distance between the SU network of 25 nodes and the PU transmitter. Increasing the PU distance results in a decrease in the average received signal to noise ratio by the SU nodes, therefore, the missed-detection rate increases with distance as shown in Fig. 14(b). Note that the false alarm rate depends on the noise level and not the signal strength, thus not shown. The results show that the proposed context-aware trust scheme performs significantly better than the context-oblivious scheme, regardless of the distance.

## VIII. Conclusion

We present a novel context-aware trust management scheme that is integrated into distributed cooperative spectrum sensing and is shown to significantly increase the resilience of the distributed cooperation to insistent spectrum sensing data falsification (ISSDF) attacks. Unlike the existing trust schemes, the proposed method enables the secondary users to perform more informed trust evaluations of their peers based on the context (whether the primary user is absent or present.) As a result our trust scheme is effective in mitigating the attackers in realistic dynamic scenarios where the primary user of the channel frequently transitions between active and inactive. We evaluate our proposed trust management scheme under Always-Yes, Always-No, and fabricating ISSDF attacks via both theoretical analysis and extensive Monte Carlo simulations. We developed a realistic model where the mobile cognitive radio ad hoc network operates in TV white space and the primary user transmitter's activity is changing over time. We show the scalability of the proposed scheme in terms of attack severity, network density and the distance of the secondary network from the primary user transmitter. Furthermore, the dynamic range of the sensitivity of the cognitive radios is shown to be considerably improved, benefiting from the proposed context-aware trust scheme.

## References

[1] 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, Jan.–Mar. 2009.

[2] S. M. Mishra, A. Sahai, and R. W. Brodersen, "Cooperative sensing among cognitive radios," in *Proc. IEEE Int. Conf. Commun.*, 2006, vol. 4, pp. 1658–1663.

[3] Z. Li, F. R. Yu, and M. Huang, "A distributed consensus-based cooperative spectrum-sensing scheme in cognitive radios," *IEEE Trans. Veh. Technol.*, vol. 59, no. 1, pp. 383–393, Jan. 2010.

[4] R. Olfati-Saber , J. Fax, and R. Murray, "Consensus and cooperation in networked multi-agent systems," *Proc. IEEE*, vol. 95, no. 1, pp. 215–233, Jan. 2007.

[5] L. Xiao, S. Boyd, and S. Lall, "A scheme for robust distributed sensor fusion based on average consensus," in *Proc. 4th Int. Symp. Inf. Process. Sensor Netw.*, 2005, pp. 63–70.

[6] D. Kempe, A. Dobra, and J. Gehrke, "Gossip-based computation of aggregate information," in *Proc. 44th Annu. IEEE Symp. Found. Comput. Sci.*, Oct. 2003, pp. 482–491.

[7] R. Chen, J.-M. Park, and K. Bian, "Robust distributed spectrum sensing in cognitive radio networks," in *Proc. 27th Conf. Comput. Commun.*, Apr. 2008, pp. 31–35.

[8] A. Vosoughi, J. R. Cavallaro, and A. Marshall, "Robust consensus-based cooperative spectrum sensing under insistent spectrum sensing data falsification attacks," in *Proc. IEEE Global Commun. Conf.*, Dec. 2015, pp. 1–6.

[9] A. Vosoughi, "Robust distributed cooperative spectrum sensing for cognitive radio ad hoc networks," Ph.D. dissertation, Rice Univ., Houston, TX, USA, May 2016.

[10] T. Qin, H. Yu, C. Leung, Z. Shen, and C. Miao, "Towards a trust aware cognitive radio architecture," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 13, no. 2, pp. 86–95, Sep. 2009.

[11] P. Kaligineedi, M. Khabbazian, and V. Bhargava, "Secure cooperative sensing techniques for cognitive radio systems," in *Proc. IEEE Int. Conf. Commun.*, May 2008, pp. 3406–3410.

[12] A. Vosoughi, J. R. Cavallaro, and A. Marshall, "Trust-aware consensus-inspired distributed cooperative spectrum sensing for cognitive radio ad hoc networks," *IEEE Trans. Cogn. Commun. Netw.*, vol. 2, no. 1, pp. 24–37, Mar. 2016.

[13] X. Liu and J. Baras, "Using trust in distributed consensus with adversaries in sensor and other networks," in *Proc. 17th Int. Conf. Inf. Fusion*, Jul. 2014, pp. 1–7.

[14] S. Althunibat, B. 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. 2015.

[15] R. Zhang, J. Zhang, Y. Zhang, and C. Zhang, "Secure crowdsourcing-based cooperative spectrum sensing," in *Proc. IEEE Conf. Comput. Commun.*, Apr. 2013, pp. 2526–2534.

[16] S. Kalamkar, P. Singh, and A. Banerjee, "Block outlier methods for malicious user detection in cooperative spectrum sensing," in *Proc. IEEE 79th Veh. Technol. Conf.*, May 2014, pp. 1–5.

[17] W. Wang, H. Li, Y. Sun, and Z. Han, "CatchIt: Detect malicious nodes in collaborative spectrum sensing," in *Proc. 28th IEEE Global Telecommun. Conf.*, 2009, pp. 5071–5076.

[18] K. Zeng, P. Paweczak, and D. Cabric, "Reputation-based cooperative spectrum sensing with trusted nodes assistance," *IEEE Commun. Lett.*, vol. 14, no. 3, pp. 226–228, Mar. 2010.

[19] N. Ahmed, D. Hadaller, and S. Keshav, "GUESS: Gossiping updates for efficient spectrum sensing," in *Proc. 1st Int. Workshop Decentralized Resour. Sharing Mobile Comput. Netw.*, New York, NY, USA, 2006, pp. 12–17.

[20] A. Vosoughi, J. Cavallaro, and A. Marshall, "A cooperative spectrum sensing scheme for cognitive radio ad hoc networks based on gossip and trust," in *Proc. IEEE Global Conf. Signal Inf. Process.*, Dec. 2014, pp. 1175–1179.

[21] W. Zhang, Z. Wang, Y. Guo, H. Liu, Y. Chen, and J. Mitola, "Distributed cooperative spectrum sensing based on weighted average consensus," in *Proc. IEEE Global Telecommun. Conf.*, Dec. 2011, pp. 1–6.

[22] W. Zhang, Y. Guo, H. Liu, Y. Chen, Z. Wang, and J. Mitola, "Distributed consensus-based weight design for cooperative spectrum sensing," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 1, pp. 54–64, Jan. 2015.

[23] D. Teguig, B. Scheers, V. L. Nir, and F. Horlin, "Consensus algorithms for distributed spectrum sensing based on goodness of fit test in cognitive radio networks," in *Proc. Int. Conf. Mil. Commun. Inf. Syst.*, May 2015, pp. 1–5.

[24] E. Yildiz, D. Acemoglu, A. E. Ozdaglar, A. Saberi, and A. Scaglione, "Discrete opinion dynamics with stubborn agents," Jan. 2011. [Online]. Available: http://ssrn.com/abstract=1744113

[25] S. Sundaram and C. Hadjicostis, "Distributed function calculation via linear iterations in the presence of malicious agents—Part I: Attacking the network," in *Proc. Amer. Control Conf.*, Jun. 2008, pp. 1350–1355.

[26] S. Liu, H. Zhu, S. Li, X. Li, C. Chen, and X. Guan, "An adaptive deviation-tolerant secure scheme for distributed cooperative spectrum sensing," in *Proc. IEEE Global Commun. Conf.*, Dec. 2012, pp. 603–608.

[27] Q. Yan, M. Li, T. Jiang, W. Lou, and Y. Hou, "Vulnerability and protection for distributed consensus-based spectrum sensing in cognitive radio networks," in *Proc. IEEE Conf. Comput. Commun.*, Mar. 2012, pp. 900–908.

[28] D. B. Johnson and D. A. Maltz, "Dynamic source routing in ad hoc wireless networks," in *Mobile Computing.* New York, NY, USA: Springer, 1996, pp. 153–181.

[29] S. Atapattu, C. Tellambura, and H. Jiang, *Energy Detection for Spectrum Sensing in Cognitive Radio*. New York, NY, USA: Springer, 2014.

[30] D. Cabric, A. Tkachenko, and R. W. Brodersen, "Experimental study of spectrum sensing based on energy detection and network cooperation," in *Proc. 1st Int. Workshop Technol. Policy Accessing Spectr.*, (TAPAS '06 series). Boston, Massachusetts, USA: ACM, 2006, Art. no. 12. doi: 10.1145/1234388.1234400.

[31] S. J. Shellhammer, "Spectrum sensing in IEEE 802.22," in *Proc. IAPR Workshop Cogn. Inf. Process.*, 2008, pp. 9–10.

[32] A. Goldsmith, *Wireless Communications*. Cambridge, U.K.: Cambridge Univ. Press, 2005.

[33] M. Gudmundson, "Correlation model for shadow fading in mobile radio systems," *Electron. Lett.*, vol. 27, no. 23, pp. 2145–2146, Nov. 1991.

[34] I. Forkel, M. Schinnenburg, and M. Ang, "Generation of two-dimensional correlated shadowing for mobile radio network simulation," in *Proc. Int. Symp. Wireless Pers. Multimedia Commun.*, Sep. 2004, vol. 21, pp. 43–47.

[35] J. Zhao, H. Zheng, and G.-H. Yang, "Distributed coordination in dynamic spectrum allocation networks," in *Proc. 1st IEEE Int. Symp. New Frontiers Dyn. Spectr. Access Netw.*, Nov. 2005, pp. 259–268.

[36] *Cognitive Wireless RAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Policies and Procedures for Operation in the TV Bands*, IEEE Standard 802.22, 2011.

**Aida Vosoughi** (S'09) received the B.S. and M.S. degrees in computer engineering from Amirkabir University of Technology, Tehran, Iran, in 2006 and 2008, respectively, the M.S. degree in electrical engineering from North Dakota State University, Fargo, ND, USA, in 2011, and the Ph.D. degree in electrical engineering from Electrical and Computer Engineering Department, Rice University, Houston, TX, USA, in 2016. From 2011 to 2016, she was a Research and Teaching Assistant in the VLSI Signal Processing Laboratory, Electrical and Computer Engineering Department, Rice University. Her research interests include security and trust management for cognitive radio ad hoc networks, very large scale integration design for wireless applications, data encryption/compression, and hardware/software codesign.

**Joseph R. Cavallaro** (S'78–M'82–SM'05–F'15) received the B.S. degree from the University of Pennsylvania, Philadelphia, PA, USA, in 1981, the M.S. degree from Princeton University, Princeton, NJ, USA, in 1982, and the Ph.D. degree from Cornell University, Ithaca, NY, USA, in 1988, all in electrical engineering. From 1981 to 1983, he was with the AT&T Bell Laboratories, Holmdel, NJ, USA. In 1988, he joined the Faculty of Rice University, Houston, TX, USA, where he is currently a Professor of electrical and computer engineering and the Director of the Center for Multimedia Communication. His research interests include computer arithmetic, and DSP, general public utilities, FPGA, and very large scale integration architectures for applications in wireless communications. During the 1996–1997 academic year, he was the Director of the Prototyping Tools and Methodology Program in the National Science Foundation. He was a Nokia Foundation Fellow and a Visiting Professor at the University of Oulu, Oulu, Finland, in 2005, and continued his affiliation there as an Adjunct Professor. He is an Associate Editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING, the IEEE SIGNAL PROCESSING LETTERS, and the *Journal of Signal Processing Systems*. He is the Chair-Elect of the IEEE CASS TC on Circuits and Systems for Communications.

**Alan Marshall** (M'88–SM'00) received the B.Sc. (Hons.) degree in microelectronic systems from the University of Ulster, Coleraine, U.K., in 1985, and the Ph.D. degree from the University of Aberdeen, Aberdeen, U.K., in 1991. He has spent more than 24 years working in telecommunications and defense industries. He was a Visiting Professor of network security at the University of Nice/CNRS, Nice, France, and an Adjunct Professor of research at the Sunway University, Subang Jaya, Malaysia. He is currently the Chair of Communications Networks at the University of Liverpool, Liverpool, U.K., where he is also the Director of the Advanced Networks Group. He formed a successful spin-out company, i.e., Traffic Observation & Management Ltd., specializing in intrusion detection and prevention for wireless networks. He is the author of more than 200 scientific papers and holder of a number of joint patents in the areas of communications and network security. His research interests include network architectures and protocols, mobile and wireless networks, network security, high-speed packet switching and quality of service and experience architectures, and distributed haptics.

Prof. Marshall is a Fellow of The Institution of Engineering and Technology. He is on the program committees of a number of IEEE conferences. He is a Section Editor (Section B: Computer and Communications Networks and Systems) of the *Computer Journal of the British Computer Society* and a Member on the editorial board of the *Journal of Networks*.

# A Context-Aware Trust Framework for Resilient Distributed Cooperative Spectrum Sensing in Dynamic Settings

Aida Vosoughi, *Student Member, IEEE*, Joseph R. Cavallaro, *Fellow, IEEE*, and Alan Marshall, *Senior Member, IEEE*

***Abstract*—Cognitive radios enable dynamic spectrum access where secondary users (SUs) are allowed to operate on the licensed spectrum bands on an opportunistic noninterference basis. Cooperation among the SUs for spectrum sensing is essential for environments with deep shadows. In this paper, we study the adverse effect of insistent spectrum sensing data falsification (ISSDF) attack on iterative distributed cooperative spectrum sensing. We show that the existing trust management schemes are not adequate in mitigating ISSDF attacks in dynamic settings where the primary user (PU) of the band frequently transitions between active and inactive states. We propose a novel context-aware distributed trust framework for cooperative spectrum sensing in mobile cognitive radio ad hoc networks (CRAHN) that effectively alleviates different types of ISSDF attacks (Always-Yes, Always-No, and fabricating) in dynamic scenarios. In the proposed framework, the SU nodes evaluate the trustworthiness of one another based on the two possible contexts in which they make observations from each other: PU absent context and PU present context. We evaluate the proposed context-aware scheme and compare it against the existing context-oblivious trust schemes using theoretical analysis and extensive simulations of realistic scenarios of mobile CRAHNs operating in TV white space. We show that in the presence of a large set of attackers (as high as 60% of the network), the proposed context-aware trust scheme successfully mitigates the attacks and satisfy the false alarm and missed-detection rates of $10^{-2}$ and lower. Moreover, we show that the proposed scheme is scalable in terms of attack severity, SU network density, and the distance of the SU network to the PU transmitter.**

***Index Terms*—Cognitive radio, context awareness, cooperative systems, mobile ad hoc networks, network security, radio spectrum management, wireless networks.**

Manuscript received December 24, 2016; revised March 24, 2017; accepted May 28, 2017. Date of publication; date of current version. This work was supported in part by the U.S. National Science Foundation under Grant ECCS-1408370, Grant CNS-1265332, and Grant ECCS-1232274, and in part by the U.S.–Ireland R&D Partnership USI033 "WiFiLoc8" grant involving Rice University (USA), University College Dublin (Ireland), and Queen's University Belfast (Northern Ireland). The review of this paper was coordinated by Dr. X. Huang. *(Corresponding author: Aida Vosoughi.)*

A. Vosoughi and J. R. Cavallaro are with the Department of Electrical and Computer Engineering, Rice University, Houston, TX 77005 USA (e-mail: vosoughi@rice.edu; cavallar@rice.edu).

A. Marshall is with the Department of Electrical Engineering and Electronics, University of Liverpool, Liverpool L69 3GJ, U.K. (e-mail: alan.marshall@liverpool.ac.uk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVT.2017.2716361

## I. Introduction

The dynamic spectrum access (DSA) paradigm, enabled by cognitive radios, facilitates flexible and efficient spectrum usage by allowing secondary users (SUs) to use licensed spectrum bands of primary users (PUs) on an opportunistic non-interference basis [1]. The SUs must perform spectrum sensing in order to avoid interference with the PUs. Cooperative spectrum sensing (CSS) that exploits the spatial diversity in the SU network effectively relaxes the sensitivity requirements on individual SUs and improves the overall sensing performance [2]. Distributed cooperative spectrum sensing (DCSS) is preferred to a centralized scheme (with a fusion center) as it is scalable, fault-tolerant and more efficient [3]. DCSS also enables cooperative sensing in cognitive radio ad hoc networks (CRAHN) where there is no base station or infrastructure. The existing DCSS schemes which are inspired by distributed average consensus algorithms are based on iterative diffusion and aggregation of data through linear iteration-based or gossip-based schemes and involve communication with direct neighbors in the network graph [4]–[6].

Spectrum Sensing Data Falsification (SSDF) [7] is a known attack for cooperative spectrum sensing schemes, where malicious SUs broadcast falsified sensing data to their neighbors in order to mislead them and compromise the spectrum sharing in the cognitive radio network. SSDF attack can cause the SUs to make incorrect decisions about the PU activity which will result in increased interference from the SUs to the PU and will also lead to underutilization of the free spectrum. Insistent SSDF (ISSDF) attack [8], [9], in particular, is aimed at iterative DCSS schemes where the attacker not only falsifies its sensing data but it also broadcasts the falsified value in every iteration of the cooperation and refrains from updating its value according to the iterative protocol. Thus, ISSDF attacks can be very harmful. Fig. 1 depicts the behavior of three main types of attackers that have been considered for CSS namely fabricating, Always-Yes, and Always-No [10], [11]. Always-Yes and Always-No attackers constantly broadcast high and low power values as their sensing reports, respectively, regardless of the PU activity state. In contrast, a fabricating attacker generates a falsified low or high value indicating the opposite of the true PU activity state.

Distributed trust schemes have been recently introduced for DCSS that require each SU node to maintain a single sliding observation vector per each SU [12], [13]. Whenever an SU node $i$ receives a value from another node $j$, node $i$ compares

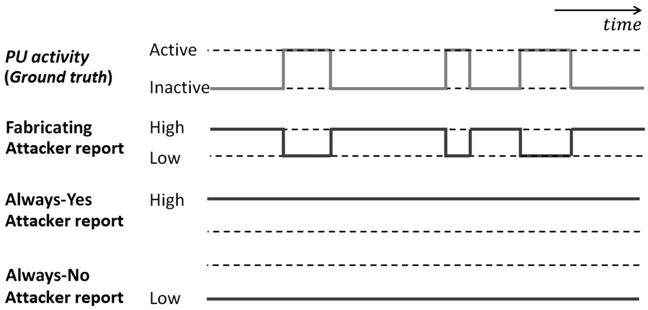



Fig. 1. PU dynamic settings and different types of attackers.

the reported value from $j$ with its own decision about the PU state. Based on this evaluation, node $i$ tags the observation from node $j$ as either an agreement or a conflict and records that in the corresponding observation vector. The trust score that node $i$ assigns to $j$ is then calculated based on the ratio of agreements over the total number of observations (the length of the observation vector) [12], [13]. We call the above trust derivation approach "context-oblivious" as the SU nodes do not distinguish between the observations based on the current PU activity context. Instead, they make blind observations and record all of the observations in a single observation vector regardless of the context.

We will show in this paper that the existing context-oblivious trust schemes are vulnerable to ISSDF attacks in dynamic settings, where the PU of the spectrum band transitions between active and inactive states over time. Thus, these techniques cannot protect the SUs and accordingly the SU nodes make incorrect detection decisions which are harmful to both the primary and secondary users of the spectrum.

Fig. 2(a) shows an example of the vulnerability of the existing agreement/conflict context-oblivious trust schemes. The Always-Yes attacker broadcasts high values (as its sensing report) all the time, even when the PU is active; therefore, in an active cycle (the duration when the PU is active), an honest node will most likely be in agreement with the Always-Yes attacker. Thus, the attacker seems to be non-malicious in the view of the honest node. As a result, the attacker is highly trusted at the end of an active cycle. Fig. 2(b) shows that in an inactive cycle, the Always-Yes attacker who has earned high trust in the previous active cycle is able to deceive the honest node to believe that the PU is active. As a result, the honest SU refrains from using the free channel. This increased false alarm rate among the honest SUs leads to no utilization or underutilization of the free spectrum which is very harmful to the SU network. The context-oblivious trust schemes have a similar vulnerability in mitigating Always-No attackers in dynamic settings, as the trust of Always-No attackers is increased in the PU inactive cycles.

In this paper, we show the vulnerability of the existing trust management schemes in dynamic settings are due to the fact that these schemes are context-oblivious. In order to solve the above-mentioned problem and to mitigate the attacks effectively, we present the following contributions:

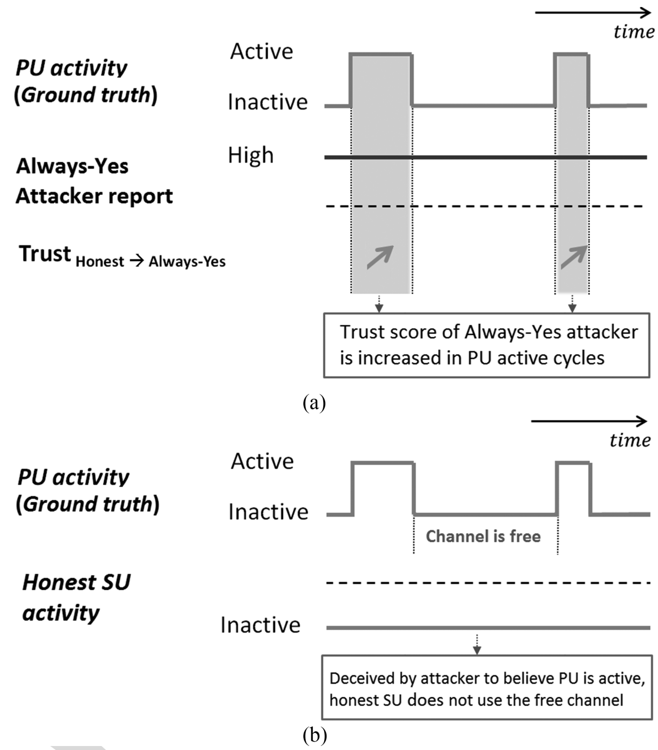



Fig. 2. An Always-Yes attack scenario in PU dynamic settings and vulnerability of the existing (context-oblivious) schemes: (a) The trust score of the Always-Yes attacker is increased when PU is active. (b) In the PU inactive cycle, the highly trusted attacker deceives the honest SU to believe PU is active; thus, the honest SU remains inactive and does not use the free channel.

1) To the best of our knowledge, this paper is the first to introduce a context-aware trust scheme for DCSS in a mobile CRAHN that is resilient to ISSDF attacks in dynamic settings where the PU frequently transitions between active and inactive states. In our proposed scheme, the trust observations are distinguished based on the speculated context: PU-Present or PU-Absent context. Thus, the trust evaluation of a peer SU is significantly more effective than the current context-oblivious schemes because it is done in a more informed manner.
2) We present a theoretical analysis to evaluate the agreement probability (thus, the level of trust) between the honest nodes and the attackers in the presence of different types of ISSDF attacks (Always-Yes, Always-No, fabricating) and considering the honest mistakes of the honest nodes. The analysis is presented for both the context-oblivious and the proposed context-aware trust schemes.
3) With both theoretical analysis and extensive Monte Carlo simulations, we show that the introduced context-aware trust scheme significantly increases the resilience of iterative DCSS schemes to ISSDF attacks in dynamic settings. Adopting the proposed trust scheme enables a mobile SU network with 20% malicious nodes in a realistic and dynamic environment to satisfy the false alarm and missed-detection rates as low as $10^{-3}$. For a similar scenario, the existing trust schemes cannot even achieve an error rate of $10^{-1}$ regardless of the detection threshold.
4) We show that our proposed trust framework is able to effectively mitigate Always-Yes, Always-No and fabricating attacks in different scenarios with high level of attack severity, even when the majority of the nodes in

the network are malicious. In addition, we show that our proposed scheme is scalable in terms of network density and the distance from the PU transmitter.

## II. Related Work

The conventional SSDF attacks and mitigation approaches against them have been well-studied in the literature for the centralized CSS schemes [7], [10], [11], [14]–[18]. A known mitigation technique against SSDF attacks is that each node assigns history-based trust scores to its neighbors and it weights their sensing reports according to the scores [7]. Recently, average consensus algorithms including gossip-based protocols and linear iteration-based schemes have been used for the DCSS applications [3], [19]–[23]. However, ISSDF attack in the iterative DCSS schemes is hardly explored.

ISSDF attackers are similar to stubborn agents [24], who have fixed opinions and do not update their beliefs based on other agents' opinions. It is shown that the initial opinion of the normal (not stubborn) agents have essentially no impact on the long-run opinion distribution [24]. Sundaram *et al.* [25] also consider a similar attack model aimed at distributed function calculation using linear iterations where the attackers do not follow the iterative update protocol and instead arbitrarily update their values in each iteration. It is shown that the network graph connectivity is a key factor in resilience to these malicious nodes [25]. However, the attack introduced in [25] is different from the ISSDF attack in that the attackers do not change (falsify) their initial values to affect the cooperation.

A trust-aware gossip-based DCSS scheme has been proposed in [20]; however, it does not consider ISSDF attacks and does not benefit from the broadcast nature of wireless and it considers sharing of binary decisions among the nodes. A proposed approach to mitigate the ISSDF attackers in iterative DCSS schemes is outlier detection [26], [27] which is based on detecting the nodes that broadcast values that are deviated from the rest of the neighbors in each iteration. However, this approach requires every node to compute a deviation threshold at each iteration which imposes a significant computational overhead on each SU. In contrast, in our proposed scheme, as will be explained, the SUs update the trust scores only once the consensus iterations are completed and therefore the computational overhead is low. Liu *et al.* [13] propose a trust scheme using trust propagation and a set of pre-trusted nodes to mitigate the effect of Byzantine adversaries in linear iterative consensus in sensor networks. However, trust propagation is costly and generally there are no pre-trusted nodes in an ad hoc network.

A distributed and low-overhead trust management scheme has been proposed recently that is integrated with a consensus-inspired DCSS scheme to mitigate ISSDF attacks [12]. However, this scheme is context-oblivious, and as explained in Section I it cannot mitigate different types of attacks in dynamic settings. To the best of our knowledge, the proposed scheme in this paper is the first context-aware trust scheme for DCSS applications that can effectively mitigate Always-Yes, Always-No and fabricating ISSDF attackers in dynamic settings without the need for centralized or pre-trusted nodes. In addition, our proposed scheme only requires the nodes to perform a single local trust evaluation per sensing round for each direct neighbor, thus the overhead is minimal.

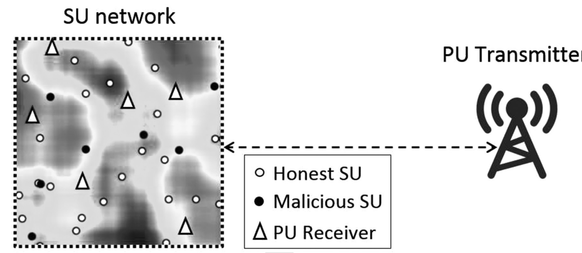



Fig. 3. System overview: Mobile SUs (honest and malicious) are moving in a square location area with diverse shadow fading. Blue represents lower received signal strength from the PU transmitter due to deep shadow fades and red represents higher signal strength.

## III. System Model

We consider a network of $n$ SU nodes that form a mobile CRAHN. The nodes are moving in a square location area within the range of a single stationary PU transmitter which is located outside the square area. Fig. 3 depicts the system overview. Random way point mobility [28] is adopted to model the SU nodes' mobility. A network of PU receivers (either mobile or stationary) may coexist with the SUs in the same location area. Therefore, whenever the PU transmitter is active, the SU's must remain silent to avoid interference to the PU receivers. The detection of a PU transmission is modeled as a binary hypothesis testing problem as follows: $H_0$ if PU is absent and $H_1$ if PU is present. Each SU is equipped with an energy detector to perform spectrum sensing by measuring the received power from the PU transmitter. The received signal by an SU can be modeled as follows:

$$y(m) = \begin{cases} w(m) & H_0 \\ s(m) + w(m) & H_1 \end{cases} \tag{1}$$

where $s(m)$ is the signal component with power $P_S$ and $w(m)$ is the zero-mean additive white Gaussian noise with noise power $P_N$. When the PU is inactive, the sensed power at an SU will essentially be equal to the received noise power. On the other hand, when the PU is active, the signal component power $P_S$ in dB can be modeled as $P_T - PL(d)[dB]$, where $P_T$ is the PU transmission power and $PL(d)$ is the path loss from the PU to the SU located in distance, $d$. If the power detector takes $M$ samples, the test statistic is given by: $\Gamma = \frac{1}{M}\sum_{m=1}^{M} y(m)y(m)^*$. Using the central limit theorem, it can be shown that for large enough $M$ [29], [30], the test statistic for a detector follows a normal distribution [31]:

$$\Gamma \sim \mathcal{N}(P_S + P_N, \frac{2(P_S + P_N)^2}{M}) \tag{2}$$

We model path loss as $PL(d) = \overline{PL}(d) + \psi_{dB}\ \ [dB]$ where $\overline{PL}(d)$ is the average path loss based on the Hata model (suburban areas variant) [32], and $\psi_{dB}$ is a Gaussian random variable in dB with zero mean and a standard deviation of $\sigma_{\psi dB}$ in dB modeling log-normal shadow fading. Therefore the total dB loss is characterized by a Gaussian distribution with mean $\overline{PL}(d)$ and standard deviation $\sigma_{\psi dB}$. The correlation between shadow fading at two locations separated by distance $\delta$ is characterized by $A(\delta) = \sigma_{\psi dB}^2 e^{-\delta/X_c}$, where $X_c$ is the decorrelation distance and is usually on the order of the size of the obstacles in the environment [32], [33]. Therefore, closely located receivers (with smaller $\delta$) experience highly correlated shadowing. We model shadows in the environment using random two-dimensional correlated shadow fading maps [34] similar to the example heatmap shown in Fig. 3.

In a non-cooperative scenario, an SU node decides on the PU activity by comparing its own received power test statistic, $\Gamma$, with a detection threshold, $\gamma$. The spectrum sensing performance is characterized by the probability of false alarm ($P_{FA}$) and missed-detection ($P_{MD}$):

$$P_{FA} = Pr(\Gamma > \gamma | H_0) \text{ and } P_{MD} = Pr(\Gamma < \gamma | H_1) \quad (3)$$

In a distributed cooperative spectrum sensing model, the SU nodes first sense and measure the received power and then share their power measurements with each other to estimate the average received power. After a number of broadcast and update iterations, each SU compares its own estimate of the average power with a threshold to make its final binary decision about the PU presence. We assume a fixed communication range for all of the SU nodes in the network. When a node broadcasts a message, all of the nodes within its predefined radius (one-hop neighbors) will receive that message. Obviously, the neighborhoods are always changing due to the mobility of the nodes; however, we assume that during one sensing period the SU network topology remains unchanged. Here we assume perfect communication between the SUs via a common control channel [35].

In a cooperative spectrum sensing model, a subset of nodes may be malicious. In this paper, we consider the insistent spectrum sensing data falsification (ISSDF) attack model [8]. ISSDF attackers broadcast falsified sensing data to their neighbors in order to cause false alarm or missed-detection errors and to deteriorate the performance of spectrum sensing at the honest (non-malicious) SU nodes. ISSDF attackers do not update their estimates according to the cooperation protocol, instead in order to make the highest impact on the network, they broadcast their falsified values in all of the iterations. We consider three types of ISSDF attackers (Always-Yes, Always-No and fabricating).

In our model, we adopt the trust-aware DCSS scheme introduced in [12]. The iterative update rule is as follows:

$$v_i(c+1) = \theta_{ii}(t) v_i(c) + \frac{\sum_{j \in R_i} \theta_{ij}(t) v_j(c)}{1 + |R_i|}, \; i = 1, ..., n \quad (4)$$

where $v_i(c)$ denotes the value at SU node $i$ at iteration $c$, and $R_i$ is the set of nodes from which node $i$ received a value in this iteration. $\theta_{ij}(t)$ denotes the trust score of node $j$ at the current sensing round $t$ in the viewpoint of node $i$ and the self-trust is $\theta_{ii}(t) = 1 - \frac{\sum_{j \in R_i} \theta_{ij}(t)}{1+|R_i|}$. The integration of trust scores as weights into the linear iteration-based consensus scheme, makes the combination biased so that the values from more trustworthy neighbors are more effective than the others. The estimation of the trust scores has been the subject of study of many of the previous research works that were mentioned in Section II and different trust schemes have been proposed [7], [10], [12], [13], [18]. In the next section, we introduce our novel distributed context-aware trust framework for trust score derivation which proves to be significantly superior to the previous methods in realistic dynamic settings.

## IV. Proposed Context-Aware Trust Framework

In a realistic cognitive radio network, the primary user of the spectrum band transitions between active and inactive states over time. We show that the dynamics of the PU activity makes the existing context-oblivious trust management schemes (e.g. [12], [13]) vulnerable to ISSDF attacks. In the existing trust schemes, each node records all of its observations from another node in a single observation vector, regardless of the context in which the observations are made.

In contrast, we introduce a context-aware trust management scheme that separates the observations based on the speculated context (PU-Absent or PU-Present). At each sensing round, each SU speculates about the PU activity using all of the available information (from its own sensing and its cooperating neighbors' reports) and conjectures the current context. Based on this speculated context, the SU will record the observations from its neighbors in the corresponding observation vectors. In future sections, we show with analysis and experiments that in realistic dynamic scenarios, our proposed context-aware trust scheme is superior to the existing context-oblivious schemes and can effectively mitigate different types of ISSDF attacks.

Next, we elaborate the proposed context-aware trust scheme. Node $i$ maintains two observation vectors per each peer node $j$: 1) "Absent observation vector", $O_{ij}^A$, 2) "Present observation vector", $O_{ij}^P$. At the end of each sensing round, node $i$ speculates and sets the context based on its own final cooperative decision: either PU-Absent or PU-Present. If at this sensing round node $i$ has received a value from node $j$, node $i$ records the observation from node $j$ based on the context. The observation is recorded in $O_{ij}^A$ if the current context set by $i$ is PU-Absent and in $O_{ij}^P$ if the context is PU-Present. The observation is binary: 0 is recorded if node $i$ and $j$ disagree and 1 is recorded if the two nodes agree on the PU activity in this sensing round. The observation vectors are essentially sliding windows of limited size, thus, if an observation vector is full at the time of recording a new observation, the oldest entry will be discarded. Algorithm 1 describes our proposed context-aware observations for context-aware trust management. $g_{ij}(t)$ denotes the initial value that node $i$ received from neighbor $j$ in the first consensus iteration of sensing round $t$, thus, referring back to (4), $g_{ij}(t)$ is equivalent to $v_j(0)$. The final estimate of node $i$ at sensing round $t$ is denoted by $y_i(t)$ which is equivalent to $v_i(c = \text{final iteration})$ in (4). $\gamma$ denotes the detection threshold.

At sensing round (time) $t$, node $i$ calculates two trust scores, $\theta_{ij}^A(t)$ and $\theta_{ij}^P(t)$ based on the absent and present observation vectors, respectively. Equation (5) shows that the scores are calculated based on the fraction of the observations that are agreements. $H(.)$ denotes the Hamming weight of the binary vector and $|.|$ is the length. The required length of the observation

**Algorithm 1:** Proposed context-aware observation for trust management. Sensing round $t$: Node $i$ observes node $j$:

**if** $(y_i(t) < \gamma)$ **then** `// i sets context: PU-Absent`
  **if** $(g_{ij}(t) < \gamma)$ **then** `// i and j in Agreement`
    $o_{ij}(t) = 1$
  **else** `// i and j in Conflict`
    $o_{ij}(t) = 0$
  **end**
  Add $o_{ij}(t)$ to $O^A_{ij}$; `// Add to Absent Vector`
**else if** $(y_i(t) > \gamma)$ **then** `// i sets context: PU-Present`
  **if** $(g_{ij}(t) > \gamma)$ **then** `// i and j in Agreement`
    $o_{ij}(t) = 1$
  **else** `// i and j in Conflict`
    $o_{ij}(t) = 0$
  **end**
  Add $o_{ij}(t)$ to $O^P_{ij}$; `// Add to Present Vector`
**end**

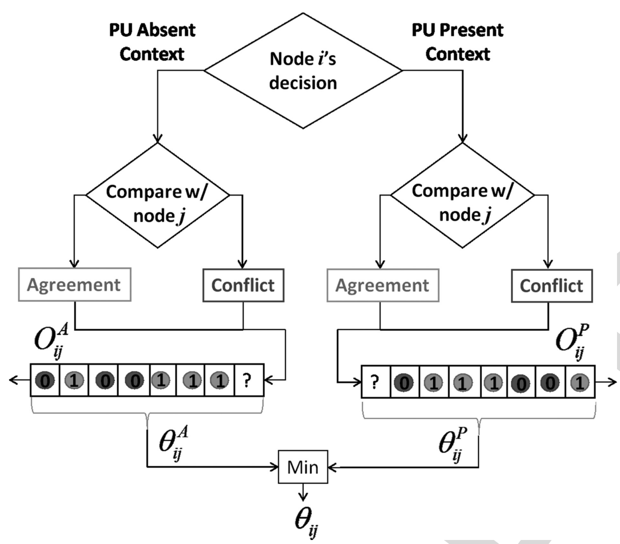



Fig. 4. Proposed context-aware trust scheme: At each sensing round, node $i$ updates the trust score assigned to node $j$ based on the minimum of the scores corresponding to the PU-Absent and PU-Present contexts.

vectors are discussed in Section IV-A in detail. We adopt the zero trust initialization strategy [12] which means the trust scores are initialized to zero and remain zero until the corresponding observation vectors are filled up to the predefined vector length. In addition, the scores are updated only when the final decisions are made at each sensing round and not in between the consensus iterations.

$$\theta^A_{ij}(t) = \frac{H(O^A_{ij})}{|O_{ij^A}|}, \ \theta^P_{ij}(t) = \frac{H(O^P_{ij})}{|O^P_{ij}|} \tag{5}$$

In each sensing round, node $i$ cannot make its final decision (and set the context) before the cooperation is complete in that round. Therefore, during the cooperation, it cannot know which of the two trust scores ($\theta^A_{ij}(t)$ or $\theta^P_{ij}(t)$) to use for a peer nodes $j$. We propose a conservative approach where the lowest of the two scores is picked as the final trust score:

$$\theta_{ij}(t) = min(\theta^A_{ij}(t), \theta^P_{ij}(t)) \tag{6}$$

Fig. 4 depicts the context-aware trust update algorithm in a flow chart representation showing the procedure of node $i$ updating the trust score assigned to node $j$ at one sensing round. Following the proposed strategy, the honest nodes take no risk and as a result, malicious nodes are always detected and excluded. The conservative score assignment strategy is advantageous because a node that is malicious in one context and not malicious in another context is always assigned a low score corresponding to the context in which it is malicious. As a result, a malicious node will have minimum effect on the honest nodes when it performs its malicious behavior.

Consider the example of an Always-Yes attacker $j$ and let us inspect how it is mitigated by an honest node $i$. Adopting the proposed scheme, all of the observations that $i$ makes from $j$ in the PU-Present context are agreements and all of the observations in the PU-Absent context are conflicts. Therefore, node $i$ perceives that node $j$ seems to be non-malicious in PU-Present context and appears to be malicious in the PU-Absent context. Since all of the observations corresponding to the PU-Absent context are conflicts, the PU-Absent context trust score is zero. Node $i$ assigns the minimum of the PU-Absent and PU-Present scores, which is zero, to $j$. As a result, node $i$ correctly detects the malicious behavior of $j$ and neutralizes its effect. Thus, separating the observations based on the context is necessary to detect the attackers. As we showed before in Fig. 2, for the same example, the context-oblivious schemes are vulnerable and ineffective.

Note that, non-malicious SUs may make honest mistakes and conjecture the context incorrectly due to shadow fading or noise (e.g., see simulation results for non-cooperative scenario in Section VII) which in turn results in an incorrect observation from a peer node. However, the properties of our proposed trust scheme helps the honest SUs to gain trust from one another and to be able to cooperate to correctly conjecture the context at each sensing round. The facilitating properties include evaluation of trust based on averaging over vectors of observations rather than an instantaneous observation and also the zero trust initialization strategy. In addition, since we take a conservative strategy for trust assignment, in case of incorrect context establishment, the malicious SUs cannot gain high trust. We consider these honest mistakes in our theoretical analysis in Section V and in our simulations. Our simulation results presented in Sections VI and VII confirm that the proposed context-aware trust scheme with conservative score assignment is significantly more stable than context-oblivious trust scheme in all of the experimented scenarios.

### A. Length of the Trust Observation Vector

Since non-malicious nodes make honest errors, instantaneous observations are not sufficient; thus, as described above, the nodes must make several observations from each other and store them in vectors and rely on the average scores. The honest nodes experience different shadowing and noise levels during time and as they move; therefore, for a sufficiently long observation vector, the average trust scores are more reliable. The shadowing characteristic (decorrelation distance or size of the shadows) and also the mobility characteristics determine the minimum required length of the observation vectors. For example, if the shadows are too large or if the nodes move very slowly, a longer vector may be needed for better trust evaluations between the nodes so that the effect of shadowing can be filtered out. On the other hand, shorter vectors may be preferred in dynamic attack scenarios to achieve fast trust update response to changes in nodes' behavior.

In conclusion, the length of the observation vectors must be determined considering the above trade-offs and the characteristics of the system. For example, as will be described later in Section VI, for our particular simulation setup, we found that the observation vector length of 8 is sufficient. As discussed before, adopting the zero trust initialization strategy, each SU initially does not trust any of the other nodes in the network. An SU can assign a non-zero trust score to another SU as soon as the observation vectors are of length 8 and some of the observations are agreements. However, the trust score of the malicious SUs will remain low because at least in one of the two contexts the conflict rate between the honest node and the attackers is high. The honest nodes then cooperate with their trusted peers to make more accurate final decisions that set the context for the future trust evaluations.

### B. *Mutual Trust Between Two Honest Nodes*

As mentioned before, the honest nodes may make non-malicious mistakes due to fading and noise; therefore, two honest nodes may not agree in their spectrum sensing decisions in a sensing round. In the case of a disagreement between two honest nodes, both of the nodes will decrease the trust score assigned to the other node. Decreasing the score of a non-malicious node that is highly unreliable and reports incorrect data to its neighbors is desired. Such a scenario occurs if there are a subset of honest nodes in the network that experience higher noise or are located in deep shadows and moving very slowly or not moving out of shadow at all. However, in a mobile network, where on average all of the nodes experience the same level of noise and shadowing and have similar mobility characteristics, the average error rates are the same for all of the peer non-malicious nodes.

Therefore, the disagreement between two non-malicious nodes is transient. As discussed before, the trust evaluation based on averaging over a vector of observations filters out these transient mistakes. As a result, over a sufficient number of observations made in both PU-Present and PU-Absent contexts every two normal honest nodes agree with each other more than they disagree. As an example for transient distrust between two honest nodes, consider an honest node $i$ that is located in a shadow area for a while and thus it incorrectly decreases the trust score of an honest neighbor $j$ since they disagree in the PU-Absent context. However, the distrust is transient because as soon as node $i$ moves out of shadow, the two nodes start to agree with each other in the PU-Absent context and $i$ increases the assigned trust score to $j$.

Certainly, there is an inevitable delay associated with the transient effect of the mutual distrust of the honest nodes and this delay will impact the resulting performance negatively. Nevertheless, this is essentially the cost that we pay for trust management to prevent the risk of potential attacks and to mitigate the malicious behavior in the cooperation. As we will show in our analysis and experiments, this negative effect is highly dominated by the positive impact of the trust scheme in detecting and excluding the malicious nodes.

Note that, although we do not explicitly present the mutual trust scores between the honest SUs in the simulation results, in all of our experiments honest nodes do assign trust scores to each other; thus, the presented missed-detection and false alarm rates do include in them the degradation due to the transient distrust. We refer the interested reader to a detailed theoretical analysis and experimental results of the honest-to-honest trust which we have presented in [9, Ch. 6].

## V. Theoretical Analysis of Context-Aware Versus Context-Oblivious Trust

As described in Section IV, a trust score that a node $k_1$ assigns to another node $k_2$ (denoted by $\theta_{k_1,k_2}$) is a measure of the probability of node $k_2$ being honest in the view of $k_1$. Node $k_1$ continuously makes observations from $k_2$ and the trust score is calculated based on the fraction of observations that are agreements. Therefore, the trust score essentially approximates the agreement probability in the most recent set of interactions between the two nodes. In this section, we analyze the agreement probability between the honest nodes and the malicious nodes for both the context-oblivious and the proposed context-aware trust schemes.

### A. *Context-Oblivious Trust Management*

In a context-oblivious trust scheme, node $k_1$ stores its observations from node $k_2$ in a single observation vector $O_{k_1,k_2}$. The event of a node $k_1$ making an observation of node $k_2$ may occur in two conditions: while PU is absent ($H_0$ is true), and while PU is present ($H_1$ is true). Therefore the probability of $k_1$ agreeing with $k_2$ can be written as:

$$\begin{aligned} Pr(agree_{k_1,k_2}) &= Pr(agree_{k_1,k_2}|H_0)Pr(H_0) \\ &\quad + Pr(agree_{k_1,k_2}|H_1)Pr(H_1) \end{aligned} \tag{7}$$

From (7), we can see that if the length of the observation vector is short relative to the PU activity period, then depending on whether $H_0$ or $H_1$ is true, one of the two components in (7) becomes dominant. For example, when PU is absent for a while, all or most of the observations in the observation vector may be from this recent PU inactive cycle and therefore the agreement between the two nodes (and consequently the trust scores) are affected almost only by the probability component corresponding to $H_0$. If the observation vector is much longer than the period of the PU activity, then on average both probability components corresponding to $H_0$ and $H_1$ will have similar effect in the trust score.

In the following paragraphs, we analyze the probability that an honest node $h_1$ agrees with a fabricating, Always-Yes or Always-No attacker. The trust scores that the honest nodes assign to their peers are essentially measured approximations of the agreement probabilities.

*1) Agreement Between Honest and Fabricating:* A fabricating attacker always reports the opposite of the truth about the PU activity. Therefore, when an honest node $h_1$ makes an observation from a fabricating attacker $f_1$, there are two conditions in which the two nodes agree: 1) when $H_0$ is true and $h_1$ makes a false alarm error, 2) if $H_1$ is true and $h_1$ makes a missed-detection error. Equation (8) shows the agreement probability between the two nodes:

$$Pr(agree_{h_1,f_1}) = Pr(F_{h_1})Pr(H_0) + Pr(M_{h_1})Pr(H_1) \quad (8)$$

where, $Pr(F_k)$ and $Pr(M_k)$ of a node $k$ denote the probability of false alarm and missed-detection of node $k$, respectively. If the cooperative decisions of the honest nodes have very low false alarm and missed-detection rates, the agreement rate with the fabricating attacker will be very small as well, thus the assigned trust scores will be small. However, when honest nodes make honest mistakes either in the presence or absence of the PU, in both cases they incorrectly agree with the fabricating attackers and as a result their associated trust scores are increased. For example if PU stays inactive for a while and the honest nodes make many false alarm errors, most of the observations in the observation vector $O_{h_1 f_1}$ are made in $H_0$ and the probability of agreement is essentially close to $Pr(F_{h_1})$ which is high. As a result, when PU finally becomes active, initially, the highly trusted fabricating attackers can significantly affect the detection performance in this cycle.

Therefore, when the context-oblivious strategy is employed, if either missed-detection or false alarm rate of the honest nodes is high, due to deep shadow or high noise, the trust score of fabricating attackers will be increased. We will discuss and show in our simulation results in the next sections that the incorrect increase in trust score of fabricating attackers due to honest mistakes has a destructive effect on the PU detection performance. In contrast, as shown later, the proposed context-aware trust scheme alleviates this problem by considering separate contexts of observations and taking the worst case (the minimum agreement among the two contexts.)

*2) Agreement Between Honest and Always-Yes:* An Always-Yes attacker always broadcasts reports that indicate the presence of the PU. Therefore, an honest node $h_1$ agrees with an Always-Yes attacker, $y_1$, in the following cases: 1) if $H_0$ is true and node $h_1$ makes a false alarm, 2) if $H_1$ is true and $h_1$ does not make a missed-detection error and actually decides that PU is present. Equation (9) derives the agreement probability:

$$Pr(agree_{h_1,y_1}) = Pr(F_{h_1})Pr(H_0) + Pr(\overline{M_{h_1}})Pr(H_1) \quad (9)$$

Obviously, when $H_1$ is true, an Always-Yes attacker's report is indeed correct. Therefore, adopting this context-oblivious trust management scheme, an honest node will incorrectly increase the trust score of an Always-Yes attacker even when the honest node has low error rate (in this case when the honest node does not make missed-detection errors). As we show later, this shortcoming of the context-oblivious trust management is significant and results in the inability of the trust scheme to mitigate Always-Yes attacks.

*3) Agreement Between Honest and Always-No:* Similarly, (10) derives the agreement probability between an honest node, $h_1$, and an Always-No attacker, $n_1$ :

$$Pr(agree_{h_1,n_1}) = Pr(\overline{F_{h_1}})Pr(H_0) + Pr(M_{h_1})Pr(H_1) \quad (10)$$

Therefore, an honest node (with a low false alarm rate) increases the trust score of an Always-No attacker when PU is absent. This makes the context-oblivious trust scheme vulnerable to Always-No attacks.

### B. The Proposed Context-Aware Trust Management Scheme

As described in Section IV, the proposed context-aware trust scheme separates the observations from each node to two contexts, PU-Absent and PU-Present. For both contexts, the event of a node $k_1$ making an observation of another node $k_2$ may occur either when $H_0$ is true or when $H_1$ is true. The context is set by $k_1$'s cooperative final decision which is its best estimate of the PU activity; therefore, "Absent observations" are not necessarily made while $H_0$ is true and "Present observations" are not necessarily made while $H_1$ is true. In this section, we analyze the agreement probability in both PU-Absent and PU-Present contexts to understand the trust scores corresponding to each of these contexts.

When a node $k_1$ makes a cooperative final decision to set the context for its observations, one of the following four events occurs:

1) $B_0^A$: $H_0$ is true and the final decision is PU-Absent. $Pr(B_0^A) = Pr(H_0)Pr(\overline{F_{k_1}})$
2) $B_0^P$: $H_0$ is true and the final decision is PU-Present. $Pr(B_0^P) = Pr(H_0)Pr(F_{k_1})$
3) $B_1^A$: $H_1$ is true and the final decision is PU-Absent. $Pr(B_1^A) = Pr(H_1)Pr(M_{k_1})$
4) $B_1^P$: $H_1$ is true and the final decision is PU-Present. $Pr(B_1^P) = Pr(H_1)Pr(\overline{M_{k_1}})$

Obviously, $B_0^P$ and $B_1^A$ occur when the node makes a false alarm and missed-detection error, respectively. In contrast, in the events $B_0^A$ and $B_1^P$, the node is not in error. We denote the event where the context is set to PU-Absent by $B^A$, which is the union of the events $B_0^A$ and $B_1^A$. Therefore, the probability of $B^A$ can be derived as follows:

$$\begin{aligned} Pr(B^A) &= Pr(B_0^A) + Pr(B_1^A) \\ &= Pr(H_0)Pr(\overline{F_{k_1}}) + Pr(H_1)Pr(M_{k_1}) \end{aligned} \quad (11)$$

Similarly, we denote the event where the context is set to PU-Present by $B^P$, which is the union of the events $B_0^P$ and $B_1^P$. Therefore, we have:

$$\begin{aligned} Pr(B^P) &= Pr(B_0^P) + Pr(B_1^P) \\ &= Pr(H_0)Pr(F_{k_1}) + Pr(H_1)Pr(\overline{M_{k_1}}) \end{aligned} \quad (12)$$

For a node $k_1$, we can derive the following conditional probabilities:

$$Pr(B_0^A|B^A) = \frac{Pr(H_0)Pr(\overline{F_{k_1}})}{Pr(H_0)Pr(\overline{F_{k_1}}) + Pr(H_1)Pr(M_{k_1})} \quad (13)$$

$$Pr(B_1^A|B^A) = \frac{Pr(H_1)Pr(M_{k_1})}{Pr(H_0)Pr(\overline{F_{k_1}}) + Pr(H_1)Pr(M_{k_1})} \quad (14)$$

$$Pr(B_0^P|B^P) = \frac{Pr(H_0)Pr(F_{k_1})}{Pr(H_0)Pr(F_{k_1}) + Pr(H_1)Pr(\overline{M_{k_1}})} \quad (15)$$

$$Pr(B_1^P|B^P) = \frac{Pr(H_1)Pr(\overline{M_{k_1}})}{Pr(H_0)Pr(F_{k_1}) + Pr(H_1)Pr(\overline{M_{k_1}})} \quad (16)$$

We denote the probability of node $k_1$ agreeing with node $k_2$ in the PU-Absent context and PU-Present context by $Pr(agree^A_{k_1,k_2})$ and $Pr(agree^P_{k_1,k_2})$, respectively. These probabilities are written in (17) and (18), respectively.

$$\begin{aligned} &Pr(agree^A_{k_1,k_2}) = \\ &Pr(agree_{k_1,k_2}|B^A) = Pr(agree_{k_1,k_2}|B_0^A)Pr(B_0^A|B^A) \\ &\qquad + Pr(agree_{k_1,k_2}|B_1^A)Pr(B_1^A|B^A) \end{aligned} \quad (17)$$

$$\begin{aligned} &Pr(agree^P_{k_1,k_2}) = \\ &Pr(agree_{k_1,k_2}|B^P) = Pr(agree_{k_1,k_2}|B_0^P)Pr(B_0^P|B^P) \\ &\qquad + Pr(agree_{k_1,k_2}|B_1^P)Pr(B_1^P|B^P) \end{aligned} \quad (18)$$

*1) Agreement Between Honest and Fabricating:*

*a) PU-absent context:* When honest node $h_1$'s final decision (and thus the context) is PU-Absent, it records its observation from a fabricating node $f_1$ in the "Absent observation vector", $O^A_{h_1,f_1}$. In this context, if $H_0$ is true (the ground truth is that PU is absent), the two nodes definitely disagree since the fabricating node's report indicates that PU is active. On the other hand, if $H_1$ is true, then the two nodes definitely agree, because the fabricating node's report indicates that PU is inactive in this case. Therefore we have the following:

$$\begin{aligned} Pr(agree_{h_1,f_1}|B_0^A) &= 0 \\ Pr(agree_{h_1,f_1}|B_1^A) &= 1 \end{aligned} \quad (19)$$

As a result by replacement in (17), the probability of agreement in the PU-Absent context is equal to the probability that $h_1$ sets the context to PU-Absent while the PU is present which means $h_1$ must make a missed-detection error. Using (14), we have the following:

$$\begin{aligned} Pr(agree^A_{h_1,f_1}) &= Pr(B_1^A|B^A) \\ &= \frac{Pr(H_1)Pr(M_{h_1})}{Pr(H_0)Pr(\overline{F_{h_1}}) + Pr(H_1)Pr(M_{h_1})} \end{aligned} \quad (20)$$

*b) PU-present context:* When honest node $h_1$'s final decision (and thus the context) is PU-Present, it records its observation from a fabricating node $f_1$ in the "Present observation vector", $O^P_{h_1,f_1}$. In this context, if $H_0$ is true, the two nodes definitely agree since the fabricating node reports that PU is active. On the other hand, if $H_1$ is true, the two nodes definitely disagree. Therefore we have the following:

$$\begin{aligned} Pr(agree_{h_1,f_1}|B_0^P) &= 1 \\ Pr(agree_{h_1,f_1}|B_1^P) &= 0 \end{aligned} \quad (21)$$

By replacement in (18), the probability of agreement in the PU-Present context is equal to the probability that $h_1$ sets the context to PU-Present while the PU is absent which means $h_1$ must make a false alarm error. Using (15), we have:

$$\begin{aligned} Pr(agree^P_{h_1,f_1}) &= Pr(B_0^P|B^P) \\ &= \frac{Pr(H_0)Pr(F_{h_1})}{Pr(H_0)Pr(F_{h_1}) + Pr(H_1)Pr(\overline{M_{h_1}})} \end{aligned} \quad (22)$$

The trust score that node $h_1$ assigns to fabricating node $f_1$ is then calculated based on the minimum of the scores derived from the two observation vectors (contexts) as described above (minimum of (20) and (22)).

*2) Agreement Between Honest and Always-Yes:* An Always-Yes attacker, $y_1$, always reports to an honest node $h_1$ that PU is active. Therefore, whenever $h_1$'s final decision is PU-Absent, the observation from $y_1$ is definitely a conflict (probability of agreement is zero) and is recorded in the "Absent observation vector". Note that, the agreement rate is exactly zero regardless of whether $h_1$ sets the context to PU-Absent by mistake (i.e., regardless of the ground truth of the PU activity.) Conversely, whenever $h_1$'s final decision is PU-Present, the observation from $y_1$ is definitely an agreement (probability of agreement is one) and is recorded in the "Present observation vector". As a result, adopting the context-aware trust, an honest node always assigns the minimum trust score which is zero to an Always-Yes attacker and thus can successfully exclude it:

$$\begin{aligned} &Pr(agree^A_{h_1,y_1}) = 0 \\ &Pr(agree^P_{h_1,y_1}) = 1 \\ &min(Pr(agree^A_{h_1,y_1}), Pr(agree^P_{h_1,y_1})) = 0 \end{aligned} \quad (23)$$

*3) Agreement Between Honest and Always-No:* An Always-No attacker, $n_1$, always indicates that PU is inactive to an honest node $h_1$. Therefore, whenever $h_1$'s final decision is PU-Absent, the observation from $y_1$ is definitely an agreement and is recorded in the "Absent observation vector". Whenever $h_1$'s final decision is PU-Present, the observation from $y_1$ is definitely a conflict and is recorded in the "Present observation vector". Taking the minimum, an honest node always assigns a zero trust score to an Always-No attacker, which means the honest node can successfully exclude the attacker:

$$\begin{aligned} &Pr(agree^A_{h_1,n_1}) = 1 \\ &Pr(agree^P_{h_1,n_1}) = 0 \\ &min(Pr(agree^A_{h_1,n_1}), Pr(agree^P_{h_1,n_1})) = 0 \end{aligned} \quad (24)$$

In the next section, we evaluate the probability of agreement between an honest node and an attacker (different types) with simulations in realistic settings. We will show that in all of the simulation scenarios under different types of attacks, adopting the context-aware trust scheme significantly reduces the effect of the attackers by assigning the lowest trust scores to them.

TABLE I
SIMULATION PARAMETERS FOR EVALUATING TRUST-AWARE DCSS

| Path Loss and Shadow Fading | | Random Way Point Model | |
|---|---|---|---|
| PU Dist. from CRAHN | 15 km | CRAHN Area | 200 m $\times$ 200 m |
| PU Antenna Height | 30 m | Min Velocity | 1 m/s |
| SU Antenna Height | 1 m | Max Velocity | 2 m/s |
| Center Freq. | 615 MHz | Min Pause | 60 s |
| Log-normal Shadowing SD ($\sigma_{\psi dB}$) | 8 dB | Max Pause | 120 s |
| Decorrelation Dist. ($X_c$) | 50 m | | |
| Transmit Power ($P_T$) | 54 dBm | | |
| **Noise and Threshold** | | **Monte Carlo Simulation** | |
| Noise Figure | 11 dB | # SU Nodes | 25 |
| Channel Bandwidth | 6 MHz | # Consensus Iter. | 4 |
| Noise Power ($P_N$) | $-95.22$ dBm | SU Node Range | 80 m |
| Threshold ($\gamma$) | $[-96, -80]$ dBm | Simulation Time | 8000 s |
| | | Sense Interval | 2 s |
| | | PU activity period | 800 s |

## VI. SIMULATION RESULTS

In this section, we present the results of our Monte Carlo simulations to evaluate the performance of our proposed context-aware trust scheme in mitigating the effect of different types of attackers. Table I describes our simulation setup. We consider a network of 25 SU nodes that are mobile in a 200 m $\times$ 200 m square location area. Each SU node can communicate with any of the other SU nodes located within its 80 m radius. We make the assumption that the sensing frequency of the SUs in the network is much faster than the PU activity frequency: Each SU senses the spectrum every 2 seconds (as recommended in IEEE 802.22 [36]) and the PU's period of activity is 800 seconds with a 50% duty cycle, which means the PU is active for 400 seconds and inactive for 400 seconds periodically. Each Monte Carlo simulation employs a different and randomly generated shadow fading map and it spans 8000 seconds during which the SUs are mobile. In each sensing round, the number of consensus iterations is 4 (See (4): the iterative update.) From the simulation parameters, it can be derived that at any point of time each of the 25 SUs in the network has 11 neighbors on average (for uniformly distributed nodes in the square location area.) Since the nodes are moving in the area, their neighborhoods are constantly changing. The presented results in this section in terms of false alarm and missed-detection performance are averaged over 10000 Monte Carlo runs to ensure sufficient randomness is captured.

As explained in Section IV-A, the minimum required length for the observation vector is determined by the characteristics of the system. According to our experimental results, the length of 8 is sufficient for our system setup and thus we have fixed $O_{\min} = 8$ in our experiments that are presented in this section (no considerable performance improvement was observed using larger observation vector lengths 16 and 32). The length of the observation vector, 8, is small compared to the period of the PU activity. In addition, for this fixed observation vector length, we experimented with smaller PU activity period of 80 s and 8 s and no noticeable difference has been observed in the performance of our proposed context-aware trust scheme. Zero trust initialization is used in all of the experiments unless otherwise stated.

### *A. Mitigating Always-Yes Attack*

Fig. 5 presents the average false alarm and missed-detection rates from Monte Carlo simulations in a scenario where 20% of the SUs are Always-Yes attackers. The figure also depicts the agreement probability between an honest SU and an Always-Yes attacker based on the analysis in Sections V-A and V-B and using average false alarm and missed-detection rates that are measured from the simulations.

Fig. 5(a) and (b) show the results corresponding to the context-oblivious and the proposed context-aware trust schemes, respectively. The context-oblivious scheme incorrectly assigns high trust scores to the Always-Yes attackers, since in the PU active cycles, the honest SUs agree with the Always-Yes nodes. In addition, for low detection thresholds, where the false alarm rate of the honest SUs is high, they agree with the Always-Yes attackers even in the PU inactive cycles. The agreement with Always-Yes attackers decreases as the threshold increases and false alarm rate decreases.

On the other hand, as discussed in Section V-B2, with the proposed context-aware scheme, an honest node is able to correctly assign the trust score of 0 to an Always-Yes attacker because it takes the minimum of the trust scores in the PU-Present and PU-Absent contexts (See (23).) In Fig. 5(b) only the minimum of the two agreement probabilities is shown which is 0. Thus, as seen from the figure, the proposed scheme effectively mitigates the attack and the false alarm rate sharply drops for the detection thresholds above the average noise power (vertical black dashed line at -95.22 dBm.) Thus, in terms of false alarm error rate, the context-aware trust strategy performs significantly better than the context-oblivious trust strategy.

In terms of missed-detection, the error rate intuitively increases for higher detection thresholds in both the context-oblivious and context-aware schemes. Since the Always-Yes attackers broadcast high values regardless of the PU activity, the malicious behavior of these attackers is advantageous when the PU is present. The reason is that the nodes in shadows might be corrected by cooperating with the Always-Yes nodes. We call this a positive side-effect of the Always-Yes malicious behavior. As a result, excluding the attackers has the counter-intuitive result of higher missed-detection errors. Since the context-oblivious trust strategy is not as effective as the context-aware scheme in mitigating Always-Yes attackers, it results in better missed-detection rate as shown in Fig. 5. Nevertheless, the negative effect of the Always-Yes attackers is significant when PU is inactive and therefore these attackers must be mitigated using the trust scheme. Receiver Operating Characteristic (ROC) curves enable us to fairly evaluate our proposed context-aware trust scheme as we need both of the missed-detection and false alarm error rates to be as small as possible at the same time for a given detection threshold. ROC curves in Fig. 6 show missed-detection and false alarm error rates for a range of detection thresholds (as described in Table I) in two scenarios: 1) in the presence of 20% Always-Yes

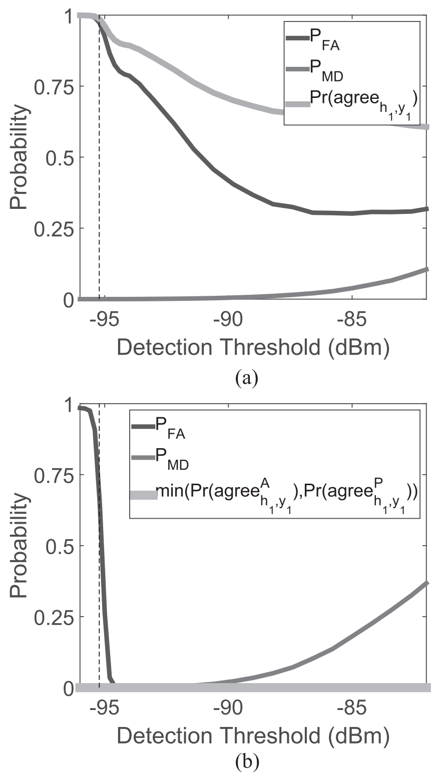



Fig. 5. 20% Always-Yes ISSDF attack (Vertical dashed lines: Noise power). (a) Context-oblivious trust management. (b) Proposed context-aware trust management.

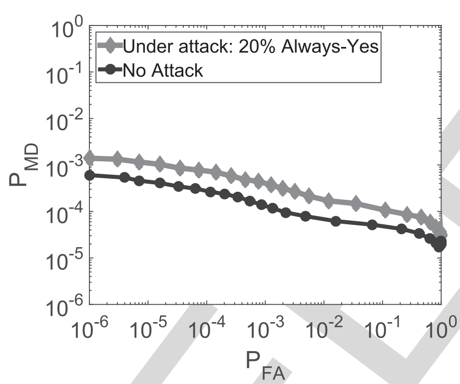



Fig. 6. ROC performance analysis: Resilient DCSS with proposed context-aware trust scheme mitigating Always-Yes ISSDF attack.

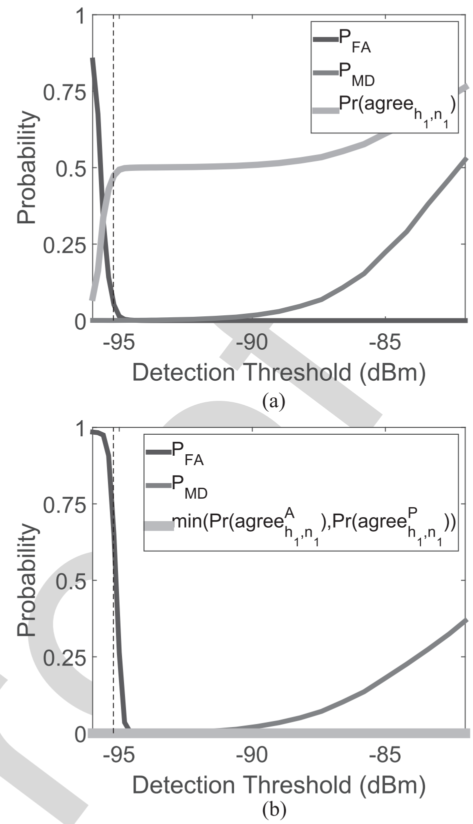



Fig. 7. 20% Always-No ISSDF attack (Vertical dashed lines: Noise power). (a) Context-oblivious trust management. (b) Proposed context-aware trust management.

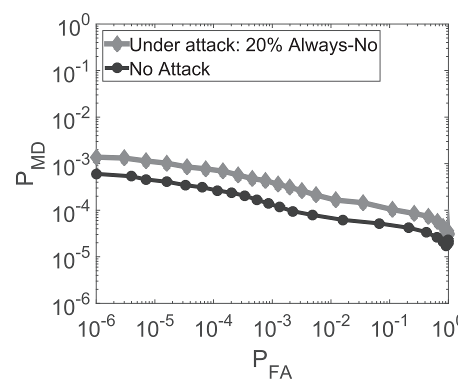



Fig. 8. ROC performance analysis: Resilient DCSS with proposed context-aware trust scheme mitigating Always-No ISSDF attack.



attackers, and 2) with no attackers. It is clear from the ROC plots that the proposed context-aware trust strategy is able to effectively contain the attack and maintain the error rate close to the no-attack case. In conclusion, as the presented ROC curve reveals, our proposed scheme offers sufficiently low missed-detection and false alarm rates at the same time in the presence of Always-Yes attackers.

## B. Mitigating Always-No Attack

Fig. 7 shows the resulting average false alarm and missed-detection rates of the Monte Carlo simulations for the scenario where 20% of the SUs conduct Always-No attacks. The figure presents the results for both context-oblivious and our proposed context-aware trust schemes. It also depicts the agreement probability based on the analysis in Sections V-A and V-B and using average false alarm and missed-detection rates from the simulations.

As can be seen from the figure, in terms of missed-detection error rate, the context-aware trust strategy performs better than the context-oblivious trust strategy. As explained in Section V-A and shown in Fig. 7(a) the context-oblivious trust incorrectly assigns high trust scores to the Always-No attackers as the agreement probability is high in all of the PU inactive cycles. For very low thresholds, where false alarm is too high, the trust score is low but as the threshold increases and the false alarm rate drops, in a PU inactive cycle, the Always-No attackers are in agreement with the honest nodes and as the duty cycle of PU is 0.5 their trust score approaches to 0.5. When the threshold is too high and the missed-detection rate starts to increase, the trust of the Always-No attackers increases even more because now the agreement between the honest nodes and the attackers also occurs in the PU active cycles. On the other hand, as seen from Fig. 7(b), with our proposed context-aware trust management, the honest nodes assign trust of zero to the Always-No attackers (See (24)) and therefore can effectively mitigate them. Similar to the case of the Always-Yes attack, here Always-No attackers have a positive side effect on the false alarm rate, meaning that since they broadcast low values even when PU is absent, they will reduce the chance of false alarms in the network. Therefore, in terms of false alarm rate, at very low thresholds (below the noise power) the context-oblivious scheme performs better than the context-aware scheme . Fig. 8 presents the resulting ROC curve for the scenario 20% Always-No attack. It is clear that the proposed context-aware trust management effectively mitigates the attackers and maintains a performance close to the no-attack scenario.

### C. Mitigating Fabricating Attack

Fig. 9 compares the performance results and the agreement probabilities of the context-oblivious and context-aware trust schemes in a scenario where 20% of the SUs are fabricating attackers. As seen from the results, the proposed context-aware trust scheme is superior to the context-oblivious trust in terms of both false alarm and missed-detection.

Fabricating attackers always broadcast a fabricated value that is the opposite of the true sensing measurement. Therefore, if an honest node does not make false alarm or missed-detection mistakes, then in both of the PU active and inactive cycles, the node will be in conflict with a fabricating attacker. However, if the honest nodes do make erroneous final decisions, then adopting the context-oblivious trust scheme, they incorrectly increase the trust of the fabricating attackers (See (8) in Section V-A.) The honest/fabricating agreement in the context-oblivious scheme shown in Fig. 9(a) confirms that for both high false alarm rate (for low thresholds on the left) and high missed-detection rate (for high thresholds on the right), the honest/fabricating agreement is increased. As a result the context-oblivious trust scheme cannot mitigate the impact of the fabricating attackers in these cases. In high false alarm case (due to high noise), the trusted fabricating attackers can increase missed-detection rate and in high missed-detection case (due to deep shadow), the trusted attackers can increase false alarm rate.

On the other hand, the proposed context-aware trust scheme, as shown in Fig. 9(b), picks the minimum of the trust scores associated with "Absent observations" and "Present observations" to filter out the mistakenly high honest/fabricating agreements at the two extremes of the threshold range. As a result, a small trust score, close to zero is assigned to the fabricating attacker. The honest nodes may be unreliable either because they are likely to make missed-detection errors (high detection thresholds relative to the signal strength) or false alarm errors (low thresholds relative to the noise level) but normally not both at the same time. Therefore, by adopting the context-aware trust strategy, the honest nodes will be able to detect the malicious behavior and to update the score of the fabricating attackers correctly. Our proposed context-aware trust management scheme is more cautious, by separating the observations in PU-Present and PU-Absent contexts and picking the minimum of the two scores (See (20) and (22).) The ROC curves in Fig. 10 clearly

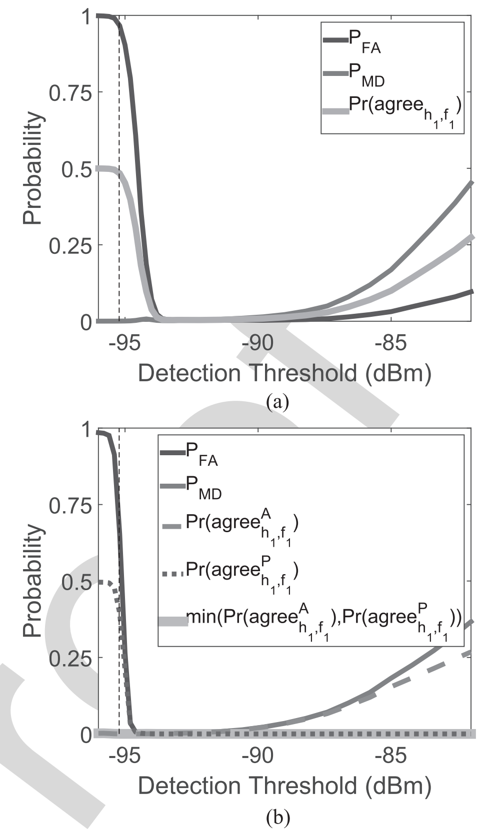



Fig. 9. 20% Fabricating ISSDF attack (Vertical dashed lines: Noise power). (a) Context-oblivious trust management. (b) Proposed context-aware trust management.

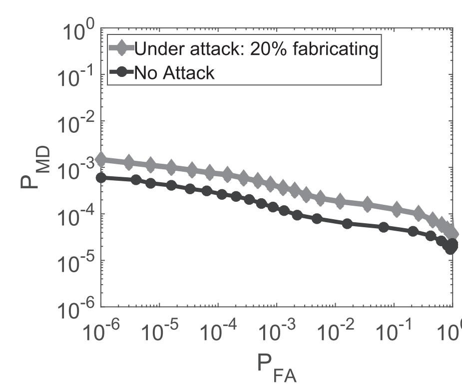



Fig. 10. ROC performance analysis: Resilient DCSS with proposed context-aware trust scheme mitigating fabricating ISSDF attack.

show that the context-aware trust is essential and effective in mitigating the attack in the case of fabricating attack as well.

### D. Discussion on the Simulation Results

The results presented in this section for various scenarios reveal that by adopting the proposed context-aware scheme, the resultant performance is consistent across all of the three types of attacks. As shown in Figs. 5, 7, and 9, unlike the context-oblivious scheme, the context-aware scheme results in the same missed-detection and false alarm rates in all of the three cases by maintaining a trust score of zero or close to zero for the attackers. Similarly, the ROC plots in Figs. 6, 8, and 10, confirm that our scheme offers essentially the same performance for all of the attack cases by successfully neutralizing the attackers (which form 20% of the network). Therefore, the proposed trust scheme offers a comprehensive solution for mitigating different attack scenarios.

In the next section, we continue our analysis and comparison with respect to different characteristics of the network including the attack severity, the SU network density and the distance of the network to the PU. In addition, we analyze the dynamic range of the detection threshold in different scenarios to satisfy a desired performance in the presence of attackers.

## VII. Comparative Performance Analysis

### A. Mitigating Attacks of Different Severity Levels

In Fig. 11 we analyze a few examples of simulation runs that show the progress over time of the average of the trust scores that the honest nodes in the network assign to one typical Always-Yes attacker. For this particular simulation, we have fixed the detection threshold to -93 dBm, a middle threshold where the average error rates for the honest nodes is not at high rates (at this threshold, the measured average probabilities of false alarm and missed-detection for an individual honest node are 0.0007 and 0.0276, respectively.)

The simulation spans 4000 sensing rounds (8000 s) and during this time the nodes are mobile. The shown plots have one data point per 50 s. For this set of experiments, we enforced an initial trust score of 0.5 for all of the nodes (rather than initializing to zero) in order to show how the honest nodes are able to reduce the trust of an attacker from 0.5 to zero and to maintain the zero trust. Fig. 11(a) shows the results where 20% of the nodes are attackers. As expected, in the context-oblivious trust scheme, the trust of the Always-Yes attacker is increased whenever PU is active. The randomness of the trust score is due to the mobility of the nodes and the changes in the neighborhoods; nevertheless, the increase in the trust score in active cycles (shaded areas) is clearly seen. As mentioned before, this is the reason why the context-oblivious trust is not effective in mitigating the attackers.

Note that, with the context-oblivious scheme, the assigned trust to the Always-Yes attacker remains high in most of the inactive cycles (white areas) showing only a small decrease. This clearly shows that once the ISSDF Always-Yes attackers in the network gained increased trust (in the active cycles), they strongly affect the final decisions of the honest nodes in the inactive cycles. Since the honest nodes mistakenly decide PU is active, the Always-Yes attackers appear to be in agreement with the honest nodes which in turn makes the honest nodes believe the attackers are trustworthy. As a result, the trust associated with the attacker is hardly decreased. In contrast, the proposed context-aware trust scheme, successfully reduces the trust of the Always-Yes attacker from the initial trust score down to 0 and keeps it low and therefore effectively excludes

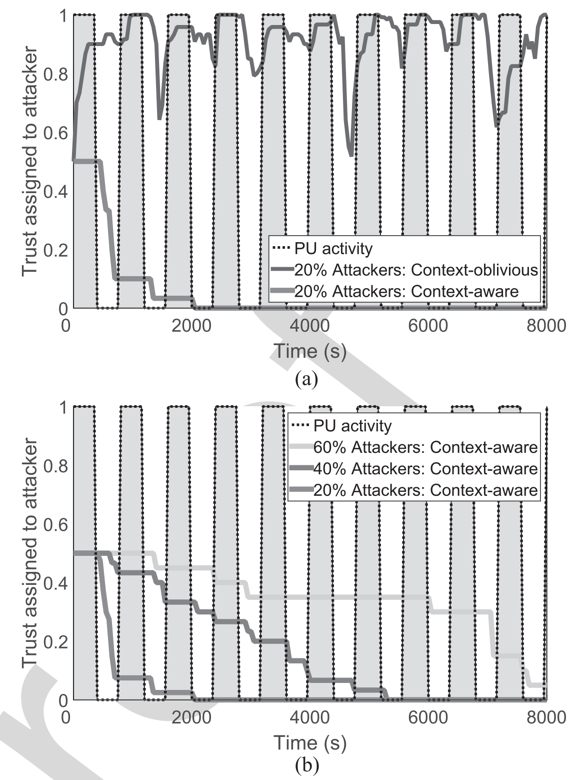



Fig. 11. Average trust score of the honest nodes assigned to a typical Always-Yes attacker. Trust scores initialized to 0.5. Detection threshold = −93 dBm. (a) 20% Always-Yes ISSDF attackers. (b) Different severities of Always-Yes ISSDF attack.

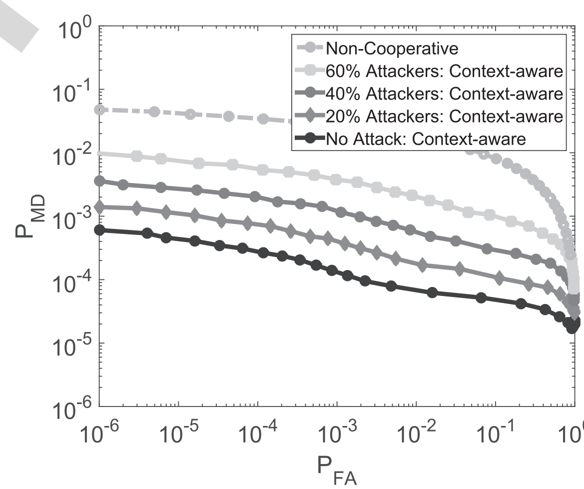



Fig. 12. ROC performance analysis: The proposed resilient DCSS scheme with context-aware trust under various Always-Yes ISSDF attack severity.

the malicious node. Fig. 11(b) compares the trust progress in different attack severity scenarios. All of the attackers in the network are of the same type (i.e., Always-Yes) and thus they strengthen each other's effect. As seen in the plot, the proposed context-aware trust successfully reduces the trust score of the attacker to zero even when the majority of the nodes are attackers (60%). The Always-Yes attackers initially have a trust score of 0.5 in the viewpoint of all of the honest nodes in the network. As the honest nodes observe these attackers, they fill up their observation vectors corresponding to both the PU-Absent and PU-Present contexts. As soon as the number of observations in a vector reaches the predefined minimum (8 observations), the trust score that is calculated based on these observations (5) replaces the initial 0.5 score.

As described in the previous section, whenever the final decision of an honest node is PU-Absent, its observation from an Always-Yes attacker will be a conflict. Therefore, as soon as 8 PU-Absent observations are made from an Always-Yes attacker, the score corresponding to PU-Absent context will be zero and thus the honest node assigns the smaller trust score of the two contexts (which is zero) to the attacker (See (23).) Although in our experiment, the PU is absent half of the time, initially due to the effect of the Always-Yes attackers (with initial trust scores of 0.5), the honest nodes are misled to decide that the PU is present most of the time. As a result, the PU-Absent observation vectors of an honest node get filled-up (i.e. reaches 8 observations) in a longer period of time compared with a no-attack scenario. The more severe the attack is, the attackers are initially more effective and it takes the honest nodes a longer time and a larger number of observations to fill their PU-Absent vectors. As a result, for more severe attacks, the convergence of the trust score towards zero takes a longer time. However, as seen in Fig. 11(b) in all of the attack scenarios including the most severe ones, eventually, the trust is reduced to zero. Therefore, the attackers are completely neutralized.

Fig. 12 shows the ROC results of Monte Carlo simulations of Always-Yes attack scenarios of different severity levels. This figure is an extension to the previously shown Fig. 6, where only 20% attack was considered. The simulation setup is the same as the setup described in Section VI (thus, adopting the zero trust initialization strategy.) The proposed scheme successfully mitigates the attacks in all of the scenarios including the case where the majority of the SUs are malicious. For comparison, we also show ROC of the non-cooperative case where the SU nodes make decisions independently without cooperation. Thus, in the non-cooperative scenario, the SU nodes are greatly affected by shadow fading and noise but they are not affected by ISSDF attackers. By utilizing the context-aware trust scheme, even under the most severe attack (i.e. 60% of the network), the resulting performance of the cooperative spectrum sensing is significantly better than the non-cooperative scenario. Therefore, the trust scheme successfully restricts the destructive effect of the attackers on the cooperation.

As shown before, the performance of the proposed trust scheme is consistent across different types of attacks. Similar results for attack severity scalability are achieved for Always-No and fabricating attacks. These results show that our proposed

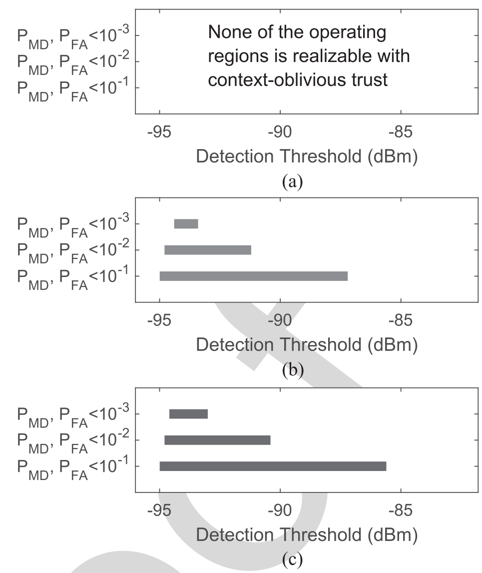



Fig. 13. Range of detection thresholds to realize desired operating regions in terms of $P_{MD}$ and $P_{FA}$. (a) Under 20% Always-Yes attack: With context-oblivious trust. (b) Under 20% Always-Yes attack: With context-aware trust. (c) No attack.

trust scheme is able to alleviate various attacks of different severity levels, thus, it provides an effective defense system against ISSDF for a wide variety of realistic scenarios.

### B. Enhanced Detection Threshold Dynamic Range

Fig. 13 compares the ranges of the detection thresholds that satisfy different operating regions in terms of missed-detection and false alarm rates for different scenarios. Under 20% Always-Yes attack, the context-aware trust helps to maintain the dynamic range of the detection threshold to approach to the honest case. In contrast, using the context-oblivious trust scheme, the attack affects the network significantly; as a result, regardless of the detection threshold, none of the operating regions, not even the most relaxed one ($10^{-1}$ error rate) can be achieved. The presented results confirm the significance of the proposed trust scheme in enhancing the flexibility and relaxing the sensitivity requirements of the cognitive radio devices.

### C. Scalability of the Proposed Trust Scheme

In this section, we analyze the scalability of the proposed context-aware trust scheme for DCSS in terms of SU network density and distance of the SU network from the PU transmitter. Fig. 14(a) shows the performance results for variable network density for a fixed detection threshold of $-94$ dBm where 20% of the nodes are fabricating attackers. In all of our experiments in the previous sections, we considered 25 SU nodes in a 200 m $\times$ 200 m location area (i.e., density of 625 SUs per km$^2$). In Fig. 14(a), however, the number of SU nodes is varied from only 5 nodes up to 50 nodes in the same area size which results in a density of 125 up to 1250 SUs per $km^2$. Therefore, we consider a variety of scenarios from a sparse to a dense SU network. In this set of simulations we use the same setup (e.g. PU

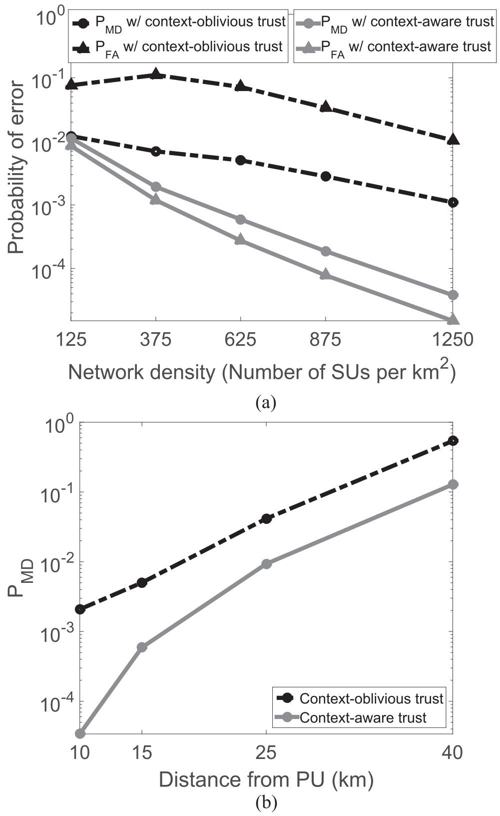



Fig. 14. Scalability analysis in terms of (a) SU network density and (b) distance from PU. Attack scenario: 20% fabricating attackers. Detection threshold = −94 dBm.

activity, SU mobility, 15 km distance from the PU transmitter) as described in Table I.

For a higher SU network density, there are more nodes in the neighborhoods and in general there is more diversity in the network that can be exploited by cooperation. As a result, both $P_{MD}$ and $P_{FA}$ should improve when the density of the network is increased. However, at the same time, in a denser network, the attackers get greater opportunity to propagate their falsified values in the network if they are not properly contained by the trust management scheme. The results presented in Fig. 14(a) shows that using the context-oblivious trust, the false alarm rate of the 375 SUs/km$^2$ case is higher than that of the 125 SUs/km$^2$ case. For denser networks, then the error rates decrease, but both false alarm and missed-detection rates remain relatively high even in the densest case. This confirms that the attackers are not mitigated adequately by the context-oblivious scheme.

In contrast, our proposed context-aware trust scheme limits the impact of the attackers and therefore, increasing the density of the nodes is beneficial as the diversity is increased. In conclusion, our proposed trust-aware DCSS scheme scales well with the network density and performs notably better than the context-oblivious trust regardless of the network density. In fact, as it is clear from Fig. 14(a), the gap between the proposed scheme and the contexts-oblivious scheme becomes more significant for denser networks.

In Fig. 14(b), we analyze the scalability in terms of the distance between the SU network of 25 nodes and the PU transmitter. Increasing the PU distance results in a decrease in the average received signal to noise ratio by the SU nodes, therefore, the missed-detection rate increases with distance as shown in Fig. 14(b). Note that the false alarm rate depends on the noise level and not the signal strength, thus not shown. The results show that the proposed context-aware trust scheme performs significantly better than the context-oblivious scheme, regardless of the distance.

## VIII. Conclusion

We present a novel context-aware trust management scheme that is integrated into distributed cooperative spectrum sensing and is shown to significantly increase the resilience of the distributed cooperation to insistent spectrum sensing data falsification (ISSDF) attacks. Unlike the existing trust schemes, the proposed method enables the secondary users to perform more informed trust evaluations of their peers based on the context (whether the primary user is absent or present.) As a result our trust scheme is effective in mitigating the attackers in realistic dynamic scenarios where the primary user of the channel frequently transitions between active and inactive. We evaluate our proposed trust management scheme under Always-Yes, Always-No, and fabricating ISSDF attacks via both theoretical analysis and extensive Monte Carlo simulations. We developed a realistic model where the mobile cognitive radio ad hoc network operates in TV white space and the primary user transmitter's activity is changing over time. We show the scalability of the proposed scheme in terms of attack severity, network density and the distance of the secondary network from the primary user transmitter. Furthermore, the dynamic range of the sensitivity of the cognitive radios is shown to be considerably improved, benefiting from the proposed context-aware trust scheme.

## References


[1] 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, Jan.–Mar. 2009.

[2] S. M. Mishra, A. Sahai, and R. W. Brodersen, "Cooperative sensing among cognitive radios," in *Proc. IEEE Int. Conf. Commun.*, 2006, vol. 4, pp. 1658–1663.

[3] Z. Li, F. R. Yu, and M. Huang, "A distributed consensus-based cooperative spectrum-sensing scheme in cognitive radios," *IEEE Trans. Veh. Technol.*, vol. 59, no. 1, pp. 383–393, Jan. 2010.

[4] R. Olfati-Saber , J. Fax, and R. Murray, "Consensus and cooperation in networked multi-agent systems," *Proc. IEEE*, vol. 95, no. 1, pp. 215–233, Jan. 2007.

[5] L. Xiao, S. Boyd, and S. Lall, "A scheme for robust distributed sensor fusion based on average consensus," in *Proc. 4th Int. Symp. Inf. Process. Sensor Netw.*, 2005, pp. 63–70.

[6] D. Kempe, A. Dobra, and J. Gehrke, "Gossip-based computation of aggregate information," in *Proc. 44th Annu. IEEE Symp. Found. Comput. Sci.*, Oct. 2003, pp. 482–491.

[7] R. Chen, J.-M. Park, and K. Bian, "Robust distributed spectrum sensing in cognitive radio networks," in *Proc. 27th Conf. Comput. Commun.*, Apr. 2008, pp. 31–35.

[8] A. Vosoughi, J. R. Cavallaro, and A. Marshall, "Robust consensus-based cooperative spectrum sensing under insistent spectrum sensing data falsification attacks," in *Proc. IEEE Global Commun. Conf.*, Dec. 2015, pp. 1–6.

[9] A. Vosoughi, "Robust distributed cooperative spectrum sensing for cognitive radio ad hoc networks," Ph.D. dissertation, Rice Univ., Houston, TX, USA, May 2016.

[10] T. Qin, H. Yu, C. Leung, Z. Shen, and C. Miao, “Towards a trust aware cognitive radio architecture,” *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 13, no. 2, pp. 86–95, Sep. 2009.

[11] P. Kaligineedi, M. Khabbazian, and V. Bhargava, “Secure cooperative sensing techniques for cognitive radio systems,” in *Proc. IEEE Int. Conf. Commun.*, May 2008, pp. 3406–3410.

[12] A. Vosoughi, J. R. Cavallaro, and A. Marshall, “Trust-aware consensus-inspired distributed cooperative spectrum sensing for cognitive radio ad hoc networks,” *IEEE Trans. Cogn. Commun. Netw.*, vol. 2, no. 1, pp. 24–37, Mar. 2016.

[13] X. Liu and J. Baras, “Using trust in distributed consensus with adversaries in sensor and other networks,” in *Proc. 17th Int. Conf. Inf. Fusion*, Jul. 2014, pp. 1–7.

[14] S. Althunibat, B. 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. 2015.

[15] R. Zhang, J. Zhang, Y. Zhang, and C. Zhang, “Secure crowdsourcing-based cooperative spectrum sensing,” in *Proc. IEEE Conf. Comput. Commun.*, Apr. 2013, pp. 2526–2534.

[16] S. Kalamkar, P. Singh, and A. Banerjee, “Block outlier methods for malicious user detection in cooperative spectrum sensing,” in *Proc. IEEE 79th Veh. Technol. Conf.*, May 2014, pp. 1–5.

[17] W. Wang, H. Li, Y. Sun, and Z. Han, “CatchIt: Detect malicious nodes in collaborative spectrum sensing,” in *Proc. 28th IEEE Global Telecommun. Conf.*, 2009, pp. 5071–5076.

[18] K. Zeng, P. Paweczak, and D. Cabric, “Reputation-based cooperative spectrum sensing with trusted nodes assistance,” *IEEE Commun. Lett.*, vol. 14, no. 3, pp. 226–228, Mar. 2010.

[19] N. Ahmed, D. Hadaller, and S. Keshav, “GUESS: Gossiping updates for efficient spectrum sensing,” in *Proc. 1st Int. Workshop Decentralized Resour. Sharing Mobile Comput. Netw.*, New York, NY, USA, 2006, pp. 12–17.

[20] A. Vosoughi, J. Cavallaro, and A. Marshall, “A cooperative spectrum sensing scheme for cognitive radio ad hoc networks based on gossip and trust,” in *Proc. IEEE Global Conf. Signal Inf. Process.*, Dec. 2014, pp. 1175–1179.

[21] W. Zhang, Z. Wang, Y. Guo, H. Liu, Y. Chen, and J. Mitola, “Distributed cooperative spectrum sensing based on weighted average consensus,” in *Proc. IEEE Global Telecommun. Conf.*, Dec. 2011, pp. 1–6.

[22] W. Zhang, Y. Guo, H. Liu, Y. Chen, Z. Wang, and J. Mitola, “Distributed consensus-based weight design for cooperative spectrum sensing,” *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 1, pp. 54–64, Jan. 2015.

[23] D. Teguig, B. Scheers, V. L. Nir, and F. Horlin, “Consensus algorithms for distributed spectrum sensing based on goodness of fit test in cognitive radio networks,” in *Proc. Int. Conf. Mil. Commun. Inf. Syst.*, May 2015, pp. 1–5.

[24] E. Yildiz, D. Acemoglu, A. E. Ozdaglar, A. Saberi, and A. Scaglione, “Discrete opinion dynamics with stubborn agents,” Jan. 2011. [Online]. Available: http://ssrn.com/abstract=1744113

[25] S. Sundaram and C. Hadjicostis, “Distributed function calculation via linear iterations in the presence of malicious agents—Part I: Attacking the network,” in *Proc. Amer. Control Conf.*, Jun. 2008, pp. 1350–1355.

[26] S. Liu, H. Zhu, S. Li, X. Li, C. Chen, and X. Guan, “An adaptive deviation-tolerant secure scheme for distributed cooperative spectrum sensing,” in *Proc. IEEE Global Commun. Conf.*, Dec. 2012, pp. 603–608.

[27] Q. Yan, M. Li, T. Jiang, W. Lou, and Y. Hou, “Vulnerability and protection for distributed consensus-based spectrum sensing in cognitive radio networks,” in *Proc. IEEE Conf. Comput. Commun.*, Mar. 2012, pp. 900–908.

[28] D. B. Johnson and D. A. Maltz, “Dynamic source routing in ad hoc wireless networks,” in *Mobile Computing.* New York, NY, USA: Springer, 1996, pp. 153–181.

[29] S. Atapattu, C. Tellambura, and H. Jiang, *Energy Detection for Spectrum Sensing in Cognitive Radio*. New York, NY, USA: Springer, 2014.

[30] D. Cabric, A. Tkachenko, and R. W. Brodersen, “Experimental study of spectrum sensing based on energy detection and network cooperation,” in *Proc. 1st Int. Workshop Technol. Policy Accessing Spectr.*, (TAPAS ’06 series). Boston, Massachusetts, USA: ACM, 2006, Art. no. 12. doi: 10.1145/1234388.1234400.

[31] S. J. Shellhammer, “Spectrum sensing in IEEE 802.22,” in *Proc. IAPR Workshop Cogn. Inf. Process.*, 2008, pp. 9–10.

[32] A. Goldsmith, *Wireless Communications*. Cambridge, U.K.: Cambridge Univ. Press, 2005.

[33] M. Gudmundson, “Correlation model for shadow fading in mobile radio systems,” *Electron. Lett.*, vol. 27, no. 23, pp. 2145–2146, Nov. 1991.

[34] I. Forkel, M. Schinnenburg, and M. Ang, “Generation of two-dimensional correlated shadowing for mobile radio network simulation,” in *Proc. Int. Symp. Wireless Pers. Multimedia Commun.*, Sep. 2004, vol. 21, pp. 43–47.

[35] J. Zhao, H. Zheng, and G.-H. Yang, “Distributed coordination in dynamic spectrum allocation networks,” in *Proc. 1st IEEE Int. Symp. New Frontiers Dyn. Spectr. Access Netw.*, Nov. 2005, pp. 259–268.

[36] *Cognitive Wireless RAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Policies and Procedures for Operation in the TV Bands*, IEEE Standard 802.22, 2011.


**Aida Vosoughi** (S’09) received the B.S. and M.S. degrees in computer engineering from Amirkabir University of Technology, Tehran, Iran, in 2006 and 2008, respectively, the M.S. degree in electrical engineering from North Dakota State University, Fargo, ND, USA, in 2011, and the Ph.D. degree in electrical engineering from Electrical and Computer Engineering Department, Rice University, Houston, TX, USA, in 2016. From 2011 to 2016, she was a Research and Teaching Assistant in the VLSI Signal Processing Laboratory, Electrical and Computer Engineering Department, Rice University. Her research interests include security and trust management for cognitive radio ad hoc networks, very large scale integration design for wireless applications, data encryption/compression, and hardware/software codesign.

**Joseph R. Cavallaro** (S’78–M’82–SM’05–F’15) received the B.S. degree from the University of Pennsylvania, Philadelphia, PA, USA, in 1981, the M.S. degree from Princeton University, Princeton, NJ, USA, in 1982, and the Ph.D. degree from Cornell University, Ithaca, NY, USA, in 1988, all in electrical engineering. From 1981 to 1983, he was with the AT&T Bell Laboratories, Holmdel, NJ, USA. In 1988, he joined the Faculty of Rice University, Houston, TX, USA, where he is currently a Professor of electrical and computer engineering and the Director of the Center for Multimedia Communication. His research interests include computer arithmetic, and DSP, general public utilities, FPGA, and very large scale integration architectures for applications in wireless communications. During the 1996–1997 academic year, he was the Director of the Prototyping Tools and Methodology Program in the National Science Foundation. He was a Nokia Foundation Fellow and a Visiting Professor at the University of Oulu, Oulu, Finland, in 2005, and continued his affiliation there as an Adjunct Professor. He is an Associate Editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING, the IEEE SIGNAL PROCESSING LETTERS, and the *Journal of Signal Processing Systems*. He is the Chair-Elect of the IEEE CASS TC on Circuits and Systems for Communications.

**Alan Marshall** (M’88–SM’00) received the B.Sc. (Hons.) degree in microelectronic systems from the University of Ulster, Coleraine, U.K., in 1985, and the Ph.D. degree from the University of Aberdeen, Aberdeen, U.K., in 1991. He has spent more than 24 years working in telecommunications and defense industries. He was a Visiting Professor of network security at the University of Nice/CNRS, Nice, France, and an Adjunct Professor of research at the Sunway University, Subang Jaya, Malaysia. He is currently the Chair of Communications Networks at the University of Liverpool, Liverpool, U.K., where he is also the Director of the Advanced Networks Group. He formed a successful spin-out company, i.e., Traffic Observation & Management Ltd., specializing in intrusion detection and prevention for wireless networks. He is the author of more than 200 scientific papers and holder of a number of joint patents in the areas of communications and network security. His research interests include network architectures and protocols, mobile and wireless networks, network security, high-speed packet switching and quality of service and experience architectures, and distributed haptics.

Prof. Marshall is a Fellow of The Institution of Engineering and Technology. He is on the program committees of a number of IEEE conferences. He is a Section Editor (Section B: Computer and Communications Networks and Systems) of the *Computer Journal of the British Computer Society* and a Member on the editorial board of the *Journal of Networks*.