Quantifying Trust for Robust Fusion While Spectrum Sharing in Distributed DSA Networks

Shameek Bhattacharjee, Saptarshi Debroy, and Mainak Chatterjee

***Abstract*—In this paper, we quantify the trustworthiness of secondary nodes that share spectrum sensing reports in a distributed dynamic spectrum access network. We propose a spatio-spectral anomaly monitoring technique that effectively captures anomalies in the spectrum sensing reports shared by individual cognitive radio nodes. Based on this, we propose an optimistic trust model for a system with a normal risk attitude and using approximation to the Beta distribution. For a more conservative and risk averse system, we propose a multinomial Dirichlet distribution-based conservative trust framework. Using a machine learning approach, we classify malicious nodes with a high degree of certainty regardless of their aggressiveness of attacks or variations introduced by the wireless environment. Subsequently, we propose two instantaneous fusion models: 1) optimistic trust-based fusion and 2) conservative trust-based fusion, which exclude untrustworthy sensing reports from participating nodes during spectrum data fusion. Our work considers random, deterministic, and preferential (ON–OFF) attack models to demonstrate the utility of our proposed model under varied attack scenarios. Through extensive simulation experiments, we show that the trust values help identify malicious nodes with a high degree of certainty.**

***Index Terms*—Dynamic spectrum access, trust and reputation, Byzantine attacks, spectrum sensing data falsification, robust fusion, secure environmental sensing capability.**

I. Introduction

RADIO spectrum allocation is typically static in nature where regulators like the Federal Communications Commission (FCC) allocate spectrum for specific services under restrictive licenses. However, recent studies have shown that most parts of the spectrum are heavily under-utilized. Dynamic spectrum access (DSA) networks allow such under-utilized bands to be used opportunistically by secondary users (i.e., non-licensees) as long as they do not cause harmful interference to the primary users (i.e., licensees). In order to detect the presence of primaries and avoid interference, the secondary users equipped with cognitive radios (CRs) undergo continuous spectrum sensing. However, due to typical wireless channel impairments like signal fading, multipath shadowing, a stand-alone radio's local sensing cannot conjecture the true occupancy status of a channel. Hence the radios, also referred to as secondary 'nodes' in this paper, participate in cooperative spectrum sensing [10], [16], where an inference on the occupancy status of a channel is made after fusing multiple local sensing results (Environmental Sensing Capability (ESC) [6]) advertised by various nodes.

However, cooperative spectrum sensing can be vulnerable when multiple malicious nodes share false local sensing reports [5]. As a result, the fused decision may be altered, hence jeopardizing the reliability of cooperative spectrum sensing. Such phenomenon where local sensing result is manipulated is known as Spectrum Sensing Data Falsification (SSDF) or Byzantine attack [4], [13]. A malicious node can advertise 'occupied' as 'available' inducing a policy violation or advertise 'available' as 'occupied' causing denial of spectrum usage. In adversarial, military, and heterogeneous cooperative sensing networks such actions are not surprising where an adversary wants to cripple the operation of others in the network using replicas that falsify [14]. Hence there is a need to evaluate the trustworthiness of nodes before considering their local spectrum sensing reports. A trust aware selection of cooperative cognitive radios is necessary to filter out spurious information (or rogue nodes) and preserve the correctness of occupancy inference.

Most of the existing approaches provide defense for centralized and infrastructure based DSA networks [13], [24], [25], [27], but solutions for distributed networks hardly exist. The common approaches are based on voting and entropy divergence which fail if malicious nodes collaborate and if there are too many of them. Some solutions require location verification which is time consuming and cannot be used in scenarios where location privacy is desired. Some discuss trust metrics without proposing how evidence for malicious behavior is gathered. Most works consider a single channel system. Hence, there is a dire need to provide a comprehensive trust framework for distributed DSA networks that works for both multi-channel collaborative and non-collaborative SSDF attacks with large number of malicious adversaries without requiring exact location of nodes.

In this paper we provide a framework for trust metrics for a distributed DSA network under SSDF attacks to: a) improve the integrity of cooperative spectrum sensing and sharing (instantaneous short term) as well as b) identify malicious

Manuscript received June 13, 2016; revised November 22, 2016 and March 20, 2017; accepted April 26, 2017. Date of publication May 8, 2017; date of current version June 16, 2017. The associate editor coordinating the review of this paper and approving it for publication was K. P. Subbalakshmi. *(Corresponding author: Saptarshi Debroy.)*

S. Bhattacharjee is with the Department of Computer Science, Missouri University of Science and Technology, Rolla, MO 65409 USA (e-mail: shameek@mst.edu.).

S. Debroy is with the Department of Computer Science, City University of New York, New York, NY 10075 USA (e-mail: saptarshi.debroy@hunter.cuny.edu).

M. Chatterjee is with the Department of Computer Science, University of Central Florida, Orlando, FL 32816 USA (e-mail: mainak@cs.ucf.edu).

Digital Object Identifier 10.1109/TCCN.2017.2702173

nodes (steady state long term). The proposed framework is context-aware and able to model trust and reputation based on the *risk attitude* of the network. To achieve this, first, we propose an anomaly monitoring technique that gathers trust evidences that could indicate the presence of anomalies in the multi-channel spectrum sensing data shared by a node's neighbors. The anomaly monitoring technique takes into account, the relative spatio-spectral orientation of the nodes' local neighborhood with respect to primary transmitters, and does not depend on the location information or any other trusted authority. Given such incomplete information about nodes' exact locations, we also demonstrate the effect of pathloss environment on the certainty of the gathered trust evidences.

Next, based on the trust evidences, we propose a *Beta expectation* based trust model that assigns trust values to neighboring nodes at different time slot. However, for a network with a higher risk attitude like a mission critical system, we also propose a *Dirichlet distribution* inspired trust model, that is able to incorporate uncertainty in the trust values. Training set results showed that Dirichlet trust is not linearly separable, and hence a linear threshold based robust classification was not possible. To circumvent this disadvantage, a Generalized Linear Model based kernel trick was employed to map Dirichlet trust into a higher dimensional plane, followed by an exponential scaling function, with trust weights bounded between -1 and +1. We propose a machine learning based classification for malicious node identification using steady state trust values. For learning the threshold for classification, we employ a supervised learning technique. For a small training network, resultant trust weights are fed to a Support Vector Machine (SVM) with known honest and malicious labels that predict a linearly separable threshold. The predicted threshold is used to classify malicious nodes for testing sets with limited a-priori knowledge.

Based on the calculated values of trust at each time, we propose an instantaneous trust based fusion that excludes the report of an untrustworthy node from participating in the cooperative spectrum fusion. Using varied and realistic simulation environments, we study the behavior under two different attack measures, viz. *Probabilistic SSDF* and *Deterministic SSDF* and then analyze which is a better attack strategy from malicious node's perspective. Our results show that the trust values of the malicious nodes are significantly lower than those which are honest. Results also show that the trust based fusion significantly outperforms the regular blind fusion performance. We compare our results with existing works and show improvement in performance, especially for high densities ($\geq$ 50%) of collaborative and aggressive malicious nodes. Finally, we consider a special case of *ON-OFF attacks* where malicious nodes use temporal preferences while launching SSDF attacks. Our results demonstrate that even for such special and challenging (to detect) attack scenarios, our proposed model successfully detects anomalous behavior and identifies malicious nodes. The salient contributions of this work are as follows:

- We provide a robust model for computing the trustworthiness of nodes that participate in cooperative spectrum sensing in a distributed DSA network under SSDF attacks. The proposed model computes trust using a received signal strength (RSS) based anomaly monitoring technique.
- We propose two trust models applicable for systems with different risk attitudes - an optimistic trust model that approximates a *Beta distribution*, and a conservative trust model using *Dirichlet distribution*.
- Our framework works for collaborative malicious nodes with (ON-OFF) or without preferential attack models under high densities of malicious nodes. Our model does not need location verification thereby obviating the possibility of location falsification.
- We propose ways to randomize attacks for a multi-channel system and show which one is better from a malicious node's perspective.
- We show that our proposed trust based fusion works better than blind fusion based prior works without the notion of trustworthiness for collaborative and non-collaborative multi-channel SSDF attacks.

The rest of the paper is organized as follows. Section II discusses the related work, and the motivation. Section III discusses the assumptions and the system model related to CR network and the adversaries. Section IV proposes a technique to gather data as trust evidence for presence of anomalies in advertised spectrum data. Section V proposes trust heuristics/models for both optimistic and conservative systems with varied risk attitudes. Sections VI and VII propose the malicious node identification and trust based fusion schemes. Section VIII discusses the simulation results for each trust models. Conclusions are drawn in the last section.

II. Related Work and Motivation

There have been a number of prior works in the defense of SSDF attacks that either concentrate on malicious node isolation or robust fusion ignoring reports from less trustworthy nodes. Rawat *et al.* [24] propose a reputation aware malicious node isolation scheme in a centralized network, where local sensing reports are sent to a central entity (fusion center) for global spectrum decisions. The authors argue that in any practical scenario majority of the nodes cannot be malicious. Hence, fusion center is bound to arrive at the correct global inference given noise is a temporal phenomenon. So global inference is matched with advertised occupancy from each node. This is known as majority voting based defense model. Other variants of this technique have been proposed in works, such as [11] and [13].

However, the majority voting based defense model and its variants assume that all nodes are inside or outside the primary's coverage area; i.e., all nodes barring wireless channel effects would arrive at the same local occupancy. It does not consider practical scenarios, where two honest nodes might legitimately have different local occupancy reports due to their *relative spatial positioning with respect to the primary transmitter*. In reality, all nodes may not legitimately sense the primaries' transmission just because few of them are placed in

a location where primary signal decays below the normalization threshold. In such a case, honest nodes may be penalized. This can result in faulty global fusion and being unfair to honest node.

Rawat *et al.* [25] propose a Kullback-Leibler (KL) divergence based method for performance analysis under collaborative SSDF attacks in a centralized DSA network. However, the authors observe that above a certain fraction of malicious nodes (50%), no reputation based fusion scheme can achieve a performance gain. They acknowledge that when malicious nodes collaboratively falsify on a particular channel and the fraction of malicious nodes exceeds 50%, KL distance and pairwise entropy techniques are not able discover malicious nodes. Hence, the KL distance method is *not robust enough for high density of collaborative malicious* nodes. As for SSDF attack defense in distributed DSA networks, the body of work meager. Moreover, existence of malicious nodes in a node's vicinity than honest nodes [4] may result in malicious nodes *easily outvoting* the honest opinions on a channel, thus, deeming such majority voting rules futile in case of distributed DSA networks.

Furthermore, most of the existing works consider centralized networks where raw received signal strength (RSS) levels are shared (soft decision) instead of binary vectors (hard decision). In [28], a consensus scheme is proposed where RSS values are shared among neighbors in a distributed DSA network. However, the authors compare the mean RSS with individual reported RSS values to exclude outliers which is the signal processing equivalent of the majority voting based defense and thus suffers similar limitations as discussed before. The soft-decision model is often bulky and hence recent literature has urged more importance towards the hard-decision models.

Some works, such as [8] consider that the location of the participating neighbor nodes are known and is used for anomaly detection. However, such assumptions are not practical due to location privacy requirements. Overall, most existing works with distributed DSA networks consider single channel system and fail to take into account the temporal aspects and nuances of SSDF attacks.

Our work in this paper, has been motivated by such limitations. In our defense model, we consider SSDF attacks in distributed DSA networks with nodes sharing multi-channel binary occupancy reports. We also address the security issues under different scenarios with varying densities of collaborative or non-collaborative malicious nodes and with different levels of aggression. Moreover, we successfully obviate the need for location information sharing among nodes without compromising the accuracy of anomaly and anomalous node detection. Finally, our proposed defense model considers topological variations and asymmetric positioning of nodes with respect to primaries' coverage, such that malicious nodes are identified with higher degrees of certainty. The optimistic and conservative models could be applied based on the required level of protection. For example, Radar Whitespaces reserved for critical military and federal operations should be a more conservative system than TV/GSM whitespaces used for civilian use. Although this work is uses a distributed network it can be applied to decentralized or centralized network as well.

TABLE I
IMPORTANT NOTATIONS

Symbol	Meaning
N	Total number of nodes
n	Total number of channels
k	Denote any particular channel
r	Sharing radius for local occupancy report
P^i	Measured power vector on n channels at node i
γ_{th}	Common threshold used to normalize power vectors
b_k	Binary Decision on a channel k, $b_k \in 0, 1$
B^i_{act}	Actual binary occupancy vector formed at a node i
B^i_{adv}	Advertised binary occupancy vector by node i
s_{T_ik}	Distance between node i and primary tower T_k for channel k
j	Set of all neighbors of i, $j \in N_i$
$P^{ij}_{predict}$	Vector of power ranges for neighbor j predicted by i
B^j_{Pre}	Binary occupancy of node j, as predicted by i
$b^j_{k\|infer}$	Predicted decision on any channel k, for B^j_{Pre}
$(\alpha, \beta, \mu)^j$	Three tuple trust evidence

III. SYSTEM AND THREAT MODELS

In this section, we discuss the system model, threat model, and assumptions for the work.

A. System Model and Assumptions

We consider a distributed DSA network deployed over a square region of a certain area, with *N* secondary nodes that undergo spectrum sensing and determine whether a channel is occupied by primaries. A secondary node *i* constructs its local occupancy vector as: $B^i_{act} = [b_1, b_2, \ldots, b_n]$, where b_k is 1 or 0 depending on whether the channel *k* is decided as occupied or unoccupied; and *n* is the total number of channels being monitored. The occupancy decision is taken by comparing the RSS measured on channel *k* with a common normalization threshold γ_{th} such that RSS more than γ_{th} denotes channel 'occupied' (b_k = 1) and vice-versa. Once the binary vector is created, secondary node *i* broadcasts this information to its neighboring node *j* within a sharing radius *r*. Similarly, *i* will also listen to broadcast messages from its neighbors. Based on the received vectors, a node employs our proposed fusion technique to obtain an estimate of spectrum usage at its location that can significantly reduce the inherent errors of spectrum sensing [13], [16]. A comprehensive list of important notations is given in Table I. Below, we outline the other assumptions:

- We assume the secondary nodes to be static and need not be aware of the geographical coordinates of other nodes. This is useful as it addresses location privacy demands.
- We assume *all nodes* transmit through a common control channel while advertising its binary vectors [10] to neighbors within a fixed radius *r*.
- We assume a primary network where each primary transmitter *whether it chooses to transmit or not*, transmits only on one channel, i.e., the channel associated with a primary transmitter is known, e.g., TV whitespace.
- We assume that the location coordinates (x_{T_k}, y_{T_k}) of a primary transmitter (T_k) transmitting on a channel (k) is known to the secondary nodes.
- We assume that there is negligible channel noise between two neighboring secondary nodes.

B. Threat Model and Assumptions

For the threat model, we only consider dishonest secondary nodes that are *malicious in nature*, i.e., nodes falsifying occupancy vector opinions on few or most of the n channels. For all practical purposes, we consider that malicious node would avoid either extremes, such as, being too aggressive that might result easy detection, or being too conservative that hardly affects the network. For analysis, we only assume independent attacks where attackers do not collaborate. However, in simulation results we will also demonstrate that our method works even if the malicious nodes are collaborative in their attacks.

We represent the level of aggression of a malicious node by *magnitude of attack* with values between 0 and 1 that is measured as the fraction of total channels where opinion is falsified. The *magnitude of attack* can be realized in two ways:

a. Deterministic Magnitude SSDF: A malicious node falsifies the report on a *fixed* number of channels every time slot. However, channels that are falsified are *randomized every time slot*. The fraction of channels falsified on every time slot is denoted by I_{attack}.

b. Probabilistic Magnitude SSDF: A malicious node falsifies report on a random number of channels every time slot, and channels falsified are also random. The nodes follow a long term mean fraction of channels that are attacked. The value of the mean is denoted as P_{attack} and it depends on how aggressive the malicious node is.

We also consider a special *ON-OFF attack model* where attack strategy is described with an ON:OFF ratio of attack and non-attack periods. Ratios with very low ON:OFF ratio signify the adversary being honest most of the time, which is not realistic.

IV. Anomaly Monitoring for Trust Evidence

To calculate trust of a node, we need to build evidence which suggests whether a node is behaving in a cooperative manner or not. This is decided by the presence or absence of anomalies in the shared binary report. We find the presence of anomalies in the advertised binary reports of a neighbor node as evidence which forms the premise for trust computation. We achieve this by *predicting the bounds on RSS* over a channel for a particular neighbor node and then apply a *normalization criterion* to obtain a predicted occupancy vector. Each node calculates a predicted occupancy vector for its neighbors. Then we *compare predicted occupancy vector with the occupancy vector that was advertised by a neighbor*. Any 'mismatch' or deviation between the predicted and advertised vectors is recorded as an event of an anomalous or non cooperative behavior. Similarly, the relative frequency of 'matches' is a measure of how much trustworthy a node's report is. Under certain conditions, a match or a mismatch decision may not be possible for a particular channel which introduces uncertainty in the evidence.

A. Predicting Bounds on Power Vector

We assume that a node i measures the power vector $P^i = \{\gamma_1^i, \gamma_2^i, \ldots, \gamma_n^i\}$, where γ_k^i is the power received on channel k and n is the total number of channels. Each node i forms its binary vector $B_{act}^i = \left[b_1^i, b_2^i, \ldots, b_n^i\right]$ from its power vector P^i by comparing γ_k^i with occupancy threshold γ_{th}, where

$$b_k^i \begin{cases} = 1 & \text{when } \gamma_k^i \geq \gamma_{th} \\ = 0 & \text{when } \gamma_k^i < \gamma_{th} \end{cases} \tag{1}$$

Each node i, advertises a public binary vector B_{adv}^i such that,

$$B_{adv}^i \begin{cases} = B_{act}^i & \text{if node } i \in H \\ \neq B_{act}^i & \text{if node } i \in M \end{cases} \tag{2}$$

where H and M denote the sets of honest and malicious nodes respectively. Just as node i advertises its binary vector to its neighbors, it also hears similar advertisements of binary occupancy vector from its neighbors. For any neighboring node $j \in N^i$, node i estimates the *bounds* on possible RSS on all channels using their mutual distance, as shown in Fig. 1. The mutual distance between the node i and its neighbor node j, can be estimated using received signal strength (RSS) lateration [2], [23]. We assume this distance between node i and its neighbor j is denoted as s_{ij}. The estimated s_{ij} could be error prone due to shadow fading and other channel impairments.

Assuming the standard propagation model for path loss and shadow fading, we can find the distance of a transmitter node given the antenna gains, transmit-receive side losses, the path loss exponent, and the standard deviation of shadow fading, when the transmitter power levels are same/known. The assumption on equal (and hence known) transmit power is reasonable as common control channel is used for broadcast messages to neighbors located within a common sharing radius. Hence using the generic model for received signal as discussed in [23] and implemented in [2]:

$$RX_{pwr} = TX_{power} + \Omega - \left(PL_{1meter} + 10log\left(d^{\omega}\right) + f_s\right) \tag{3}$$

where, RX_{pwr} is the observed (detected) received signal power when a potential neighbor transmits with TX_{power}, Ω abstracts all antenna gains, d is the distance between transmitter and receiver which is unknown; f_s is the shadow fading between two secondary nodes; PL_{1meter} is the near field reference power, and ω is the path loss exponent. The deductive portion of Eqn. (3) is given by the path loss such that:

$$PL_{1meter} + 10log\left(d^{\omega}\right) + f_s = PL_{Tx-Rx} \tag{4}$$

The shadow fading component on the dB scale is a normal distribution with zero mean and standard deviation of shadow fading such that $f_s = \mathcal{N}(0, \sigma_s)$, where σ_s can be derived through empirical studies. Given TX_{power}, Ω, RX_{pwr}, PL_{1meter}, ω and σ_s for the concerned region, we calculate the estimated distance d from Eqn. (3). This distance is the estimated distance between node i and its neighbor j and is denoted as s_{ij} such that $d = s_{ij}$ and is given by:

$$d = 10^{\frac{TX_{pwr} - RX_{pwr} + \Omega + f_s - PL_{1meter}}{10\omega}} = s_{ij} \tag{5}$$

Calculated s_{ij} varies with the variation of f_s which follow a normal distribution with mean 0 dB and a non-zero standard deviation of σ_s dB. The standard deviation of shadow fading (in dB) can range from 3 to 7 dB in certain indoor environments [2], and as high as 8 to 12 dB in certain outdoor to indoor environments [1]. The path loss exponent ω is also

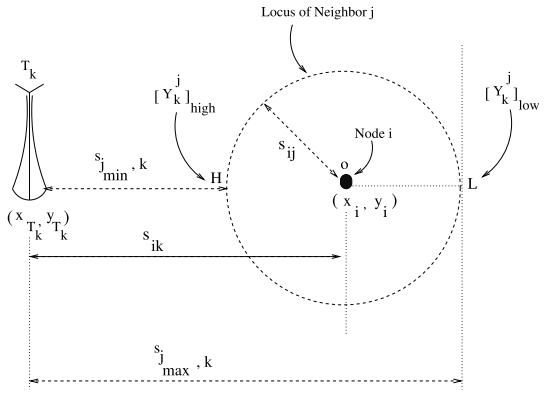


Fig. 1. Maximum and minimum RSS on channel k of node j.

heavily dependent on the type of physical environment and is typically greater than 2 in environments where obstructions are present. Typical value for an indoor office environment may be 3.5, a dense commercial or industrial environment 3.7 to 4.0, and a dense home environment might be as high as 4.5. We have used these realistic values in our simulations for validation.

The distance s_{ij} allows us to draw a circle of radius of s_{ij} around the monitoring node i. This circle is the locus of the neighboring node j's location which can be anywhere on this circle. We draw a straight line from the center of the circle to the primary transmitter T_k located at (x_{T_k}, y_{T_k}) as shown in Fig. 1. Under ideal conditions, the RSS due to T_k will be maximum on the point of the circle that is closest to T_k (point H and distance $s_{j_{min},k}$) and minimum at the point farthest from T_k (point L and distance $s_{j_{max},k}$). We denote the RSS values at these two locations as $[\gamma_k^j]_{high}$ and $[\gamma_k^j]_{low}$ respectively. For all other locations within the circle, the RSS varies between $[\gamma_k^j]_{high}$ and $[\gamma_k^j]_{low}$.

Modeling the primary signal propagation considering fading over long distances, we have

$$\gamma_k^i = P_k \times \frac{A^2}{s_{ik}^{\omega}} + f_l; \tag{6}$$

where γ_k^i is the RSS detected on channel k at node i for a primary transmitter T_k, A is the frequency constant, s_{ik} is the distance between primary tower T_k and node i, and P_k is the transmit power of T_k, $A = \frac{\lambda}{4\pi}$ where λ is wavelength of light, f_l is shadow fading factor over long distances between primary towers and secondary nodes such that $f_l = \mathcal{N}(0, \sigma_l)$. f_s and f_l are different because the extent of shadow fading is different between two secondary nodes and between a primary and secondary, and usually $\sigma_s < \sigma_l$. From Eqn. (6), we get P_k which is used to calculate the bounds on possible received power due to the primary's transmission (see Fig. 1) as:

$$\left[\gamma_k^j\right]_{high} = P_k \times \frac{A^2}{s_{j_{min},k}^{\omega}} + f_l; \tag{7}$$

$$\left[\gamma_k^j\right]_{low} = P_k \times \frac{A^2}{s_{j_{max},k}^{\omega}} + f_l; \tag{8}$$

Now we divide the Eqn. (6) by Eqns. (7) and (8) to calculate $[\gamma_k^j]_{high}$ and $[\gamma_k^j]_{low}$, since s_{ik}, $s_{j_{min},k}$ and γ_k^i are known to node i. Thus the predicted RSS of node j is a 2-tuple vector $P_{predict}^{ij} = \left[([\gamma_1^j]_{low}, [\gamma_1^j]_{high}), ([\gamma_2^j]_{low}, [\gamma_2^j]_{high}), \ldots, ([\gamma_n^j]_{low}, [\gamma_n^j]_{high})\right]$.

B. Normalization and Trust Evidence Formation

With the estimated RSS known, the occupancy inferred by node i about node j on channel k is derived as:

$$b_k^j|_{infer} = \begin{cases} 0 & \text{if both } \left[\gamma_k^j\right]_{low} \text{ and } \left[\gamma_k^j\right]_{high} \leq \gamma_{th}; \\ 1 & \text{if both } \left[\gamma_k^j\right]_{low} \text{ and } \left[\gamma_k^j\right]_{high} \geq \gamma_{th}; \\ X & \text{otherwise} \end{cases} \tag{9}$$

where X denotes that no inference could be drawn. The overall predicted occupancy vector, given the mutual distance between node i and j, is given as:

$$B_{pre}^j = \left[b_1^j|_{infer}, \ldots, b_n^j|_{infer}\right]; \quad b_k^j|_{infer} \in 0, 1, X \tag{10}$$

Next, node i compares predicted B_{pre}^j with received (from j as advertised) $B_{adv}^j = [b_1^j, \ldots, b_k^j..b_n^j]$ for each channel and records the results using criterion in Eqn. (11). The 'mismatches' (denoted by β), and 'matches' (denoted as φ) are recorded, and channels with value X in are considered 'undecided' and recorded as μ. If Q_k^j is the overall comparison outcome, then:

$$Q_k^j = \begin{cases} \varphi^j & \text{if } b_k^j|_{infer} = b_k^j; \\ \beta^j & \text{if } b_k^j|_{infer} \neq b_k^j; \\ \mu^j & \text{otherwise} \end{cases} \tag{11}$$

The total number of matches, mismatches and undecided for each node j is denoted as η_{φ_j}, η_{β_j} and η_{μ_j} such that $\eta_{\varphi_j} + \eta_{\beta_j} + \eta_{\mu_j} = n$. This 3 tuple vector forms the trust evidence. The φ_j is a treated as a positive rating, β_j is a negative rating, and μ_j is a neutral or uncertain rating. The more the number of positive ratings relative to the overall number of ratings, more is the positive behavior and vice-versa. The number of neutral ratings increases or decreases confidence on our estimates which is discussed in the next section. The complexity of this method is $O(mn)$, where m is the average number of neighbors of a node. In a distributed DSA network, this complexity is manageable as $m << N$ with the sharing region being very small compared to the deployment area.

V. Trust Models

In this section, we propose two models, viz., optimistic and conservative that use number of matches, mismatches, and undecided variables to compute and manage node trust in a distributed environment with malicious nodes.

A. Beta Distribution Based Optimistic Model

Given ternary evidence, we need to map the eventual decision on whether to trust or not. Since more number of matches would intuitively denote more trustworthiness, we seek to model trustworthiness as a relative frequency of matches to the total number of possible outcomes. To account for the number of undecided η_{μ^j}, we split η_{μ^j} into the ratio η_{φ^j}:η_{β^j}

TABLE II
TRUST-CERTAINTY TUPLE; N=40

Scenario	φ	β	μ	Trust,Certainty	Beta Trust
1	14	13	13	$0.51, 0.675$	0.50
2	19	18	3	$0.51, 0.925$	0.50
3	22	0	18	$1.00, 0.55$	0.97

and add the corresponding fraction $\frac{\eta_{\varphi^j}}{\eta_{\varphi^j}+\eta_{\beta^j}}\eta_{\mu^j}$ to η_{φ^j} in order to generate the relative frequency of matches. We use this split ratio as the attacks did not have any preference over the undecided channels X, i.e., the attacks were uniformly random over the channels.

Thus the proportion of matches is updated as $\eta_{\varphi^j}+\frac{\eta_{\mu^j}}{\eta_{\varphi^j}+\eta_{\beta^j}}\times \eta_{\varphi^j}$. The proportion or relative frequency of matches to the total number of channels can be treated as the instantaneous trust value for node j as computed by node i and is given by

$$E^{j,i} = \frac{\eta_{\varphi^j} + \frac{\eta_{\mu^j}\eta_{\varphi^j}}{\eta_{\varphi^j}+\eta_{\beta^j}}}{\eta_{\varphi^j}+\eta_{\beta^j}+\eta_{\mu^j}} \tag{12}$$

where $0 \leq E^{j,i} \leq 1$. Values of $E^{j,i}$ closer to 1 indicate more trustworthiness.

1) Bounds on Trust Values: By computing the bounds over the trust value from Eqn. (12), we provide a *certainty* measure that express how much confidence we have over the calculated value. This becomes particularly important to distinguish scenarios with high and low number of undecided. The number of μ^j's can have any number of matches or mismatches which is unknown to the monitoring node. The trust attains a *maximum value if all* μ^j's *are matches* and a *minimum value when all* μ^j's *are mismatches*, i.e., $E^{j,i}_{high} = \frac{\eta_{\varphi_j}+\eta_{\mu_j}}{N}$ and $E^{j,i}_{low} = \frac{\eta_{\varphi_j}}{N}$ respectively for the maximum and minimum cases. The interval $[E^{j,i}_{low}, E^{j,i}_{high}]$ depends on how large μ^j is. The larger this interval the lesser the probability of the true relative frequency to be closer to the expected (trust) value. Under uniform attacks, the extreme cases of all undecided being either all matches or all mismatches is low, as there is no preference over the channels attacked.

2) Certainty Over Trust Value: We argue that larger the range $\delta = (E^{j,i}_{high} - E^{j,i}_{low})$, lesser should be the confidence. Hence we use $1-\delta$ as the metric that defines how much confident or certain we are about $E^{j,i}$. In cases where number of undecided are less, there are more known matches or mismatches which makes the trust computation more certain. Hence, when i assigns a trust value to a neighbor j where perfect information is not present, it uses metric called *certainty* to indicate the confidence over $E^{j,i}$. The certainty is defined as $a^{j,i} = 1-\delta$. The trust-certainty tuple for neighbor j is represented as $(E^{j,i}, a^{j,i})$. Higher $a^{j,i}$ indicates higher confidence on the computed trust $E^{j,i}$ value.

For example, consider the three scenarios shown in Table II. The first two scenarios have the same trust but the node in Scenario 2 has more certainty, because true observations are known on 37 out of the 40 channels. Thus the trust value of scenario 2 will have more confidence than in scenario 1. Similarly, scenario 3 is the least trustworthy followed by 2 and 1.

3) Trust Evidence Coarsening: Now the proposed optimistic trust heuristic does not match with any known distribution, hence it lacks mathematical tractability, easy calculation of higher moments, and 95% confidence intervals. Additionally, it cannot be used in Bayesian systems where update of parameters are based on incremental evidence due to violation of Cromwell's rule. Hence, we propose an approximation of the optimistic heuristic with the well-known Beta distribution that is widely used for trust modeling. Ternary evidences can be alternatively modeled by coarsening it into a binary space [20] to make it mathematically tractable with Beta distribution [17]. Given that r is the number of positive and s is the number of negative outcomes, the trust is given by the mean of beta distribution with parameters specified by $\alpha = r+1$ and $\beta = s+1$. The mean of the pdf in Eqn. (13) can accurately model trust metrics as:

$$E(p) = \frac{\alpha}{\alpha+\beta} = \frac{r+1}{r+s+2} \tag{13}$$

Following this, we can treat the floor of the numerator in Eqn. (12) as the coarsened number of matches denoted as α^c_j and $N-\alpha_j = \beta^c_j$ as the coarsened number of mismatches. Given this, the trust value modeled as the expectation of a beta distribution with parameters $(\alpha^c_j+1, \beta^c_j+1)$ is expressed as:

$$E^{j,i}_{beta} = \frac{\alpha^c_j+1}{\alpha^c_j+\beta^c_j+2} \tag{14}$$

where $\alpha^c_j = \lfloor \eta_{\varphi_j} + \frac{\eta_{\mu_j}\eta_{\varphi_j}}{\eta_{\varphi_j}+\eta_{\beta_j}} \rfloor$. We can observe from Table II, that Eqns. (12) and (14) almost give the same value. Hence we say that modified beta expectation based trust values approximates our optimistic trust heuristic. This observation holds true as we assume the channels chosen for attack are uniformly random.

4) Analysis of Error Bounds: Suppose P_{fa} and P_{md} are the probabilities of missed detections and false alarms per channel. Let the expected number of unwarranted mismatches between two honest nodes caused by errors be given by β_{error}. Similarly, α_{error} is the unwarranted matches which were actually mismatches. Let the P_I and P_B be the probabilities of a channel to be idle or busy.

The probability that two honest nodes will legitimately have a different opinion about a channel due to missed detections and false alarms is

$$\beta_{error} = N(2P_I P_{fa}(1-P_{fa}) + 2P_B P_{md}(1-P_{md})) \tag{15}$$

Hence $\lceil \beta \rceil_{error}$ number of mismatches on average could be caused by errors, assuming each of the N channels have identical properties. Hence, β_{error} mismatches may be discounted and does not amount to malicious behavior. The calculations of P_I, P_B, P_f and P_m have been already studied extensively in the existing literature.

The other error metric α_{error} can be calculated as:

$$\begin{aligned}\alpha_{error} = N\Big[1 - \Big\{ & P_I P_{fa}(1-P_{fa})(1-P_{attack}) + P_I P^2_{fa} P_{attack} \\ & + P_I(1-P_{fa})P_{fa}(1-P_{attack}) + P_I(1-P_{fa})^2 P_{attack} \\ & + P_B P_{md}(1-P_{md})(1-P_{attack}) + P_B P^2_{md} P_{attack} \\ & + P_B(1-P_{md})P_{md}(1-P_{attack}) \\ & + P_B(1-P_{md})^2 P_{attack} \Big\}\Big]\end{aligned}$$

Hence, the net error in terms of matches and mismatches is $\pm(\beta_{error} - \alpha_{error})$ for malicious nodes. This net error divided by number of channels N is the approximate error in the trust metrics which is the confidence interval over trust values. For honest nodes, α_{error} is very small, because $P_{attack} = 0$.

B. Dirichlet Expectation Based Conservative Model

Through splitting the 'undecided' in the ratio of observed matches and mismatches, we get to a trust metric which models behavior. However, such a split can be argued against under scenarios like non-uniform or pseudo-random channel preference of adversaries, high number of uncertain ratings etc. Particularly coarsened binomial models cannot distinguish between cases with very large and very small number of uncertain ratings. Hence the use of a multinomial model for trust modeling such as the Dirichlet distribution is required, which is the multivariate generalization of the corresponding binomial models.

Multinomial distribution is the generalization of the binomial distribution with $z > 2$ possible outcomes where each trial results in one out of z outcomes from a set of N possible trials. We can model match, mismatch and undecided as the possible outcomes on the inference over each channel; the total number of channels being N and hence $z = 3$. Thus observation counts from the trust evidence fits very well with concept of multinomial distribution. Given this, observation for any node can be treated as multinomial distribution given the probabilities of occurrence of each outcome.

General theory of Dirichlet distribution says that, if $x_1, \ldots x_i, \ldots x_z$ are the unknown probabilities associated with z events, and the evidence is d_l, for the l-th event, then the posterior degree of belief on each x_l having accounted for evidence parameter d_l is given as $p(x_l|d_l) = \frac{p(d_l|x_l)p(x_l)}{p(d)}$. The evidence parameter d_l, is defined as $d_l = r_l + Ca_l$, where r_l represent the most recent count for event l and a_l represents a prior base rate and C represents an a-priori constant whose value depends on whether assumed prior is informative or not [18].

The above posterior $p(x_l|d_l)$ can be calculated using the posterior Dirichlet multinomial distribution function with variables $\vec{x} = (x_1, x_2, \ldots, x_z)$ and parameters $\vec{d} = (d_1, d_2, \ldots, d_z)$ is defined as:

$$f(\vec{x}|\vec{d}) = \frac{\Gamma\left(\sum_{l=1}^{z} d_l\right)}{\prod_{l=1}^{z} \Gamma(d_l)} \prod_{l=1}^{z} x_l^{d_l - 1}, \tag{16}$$

where $x_1, x_2, \ldots, x_z > 0$, $\sum_{l=1}^{z} x_z = 1$, $d_1, \ldots, d_z > 0$. The relation between observation parameter d_l and actually observed outcome frequency r_l where $\sum_{l=1}^{z} a_l = 1$ and $C > 0, a_l > 0$ such that zero occurrence of an outcome preserves the condition that $d_l > 0$. Since trust is an expectation of positive behavior [15], the trust is given by the mean vector for Eqn. (16) and is given as

$$E(x_l|\vec{d}) = \frac{d_l}{\sum_{l=1}^{z} d_l} \tag{17}$$

The degrees of belief associated with the outcomes are expressed as the mean of each outcome.

1) Applying Dirichlet Model to Trust Evidence: For our scenario, the most recent observation vector is the multinomial trust evidence $\boldsymbol{r} = \{\eta_\varphi, \eta_\beta, \eta_\mu\}$. Thus the data parameter is defined as:, $d_1 = \eta_\varphi + Ca(x_1)$, $d_2 = \eta_\beta + Ca(x_2)$ and $d_3 = \eta_\mu + Ca(x_3)$. Since before trust establishment, there is no reason to believe a node has a particular pre-disposition to behave in a positive, negative or uncertain way, we assume a uniformly distributed non-informative prior. Since there are 3 outcomes, the prior initial base rate is given by $a(x_l) = \frac{1}{3}$ and is set as $C = 3$. Given this $d_1 = \eta_\varphi + 1$; $d_2 = \eta_\beta + 1$; $d_3 = \eta_\mu + 1$. Now that we have the parameters of the Dirichlet distribution, we can express the expected degrees of belief associated with the events of match, mismatch and undecided in terms of the observed trust evidence using Dirichlet distribution as:

$$E_\varphi = \frac{\eta_\varphi + 1}{\eta_\varphi + 1 + \eta_\beta + 1 + \eta_\mu + 1} \tag{18}$$

Similarly, $E_\beta = \frac{\eta_\beta + 1}{\eta_\varphi + \eta_\beta + \eta_\mu + 3}$ and $E_\mu = \frac{\eta_\mu + 1}{\eta_\varphi + \eta_\beta + \eta_\mu + 3}$. Hence for each node j, we have $E_\varphi = E^b_{ji}$ representing degree of belief, $E_\beta = E^d_{ji}$ representing degree of disbelief and $E_\mu = E^u_{ji}$ reflecting degree of uncertainty of node j based on gathered trust evidence of node i from the anomaly monitoring phase.

2) Interpreting Belief Using Subjective Logic Theory: The proposition that a node will cooperate can either be true or false and hence is a binary proposition. However, due to inherent uncertainty and imperfect knowledge caused by lack of evidence, it is not possible to infer with certainty that the proposition is true or false. Hence an *opinion* is given about the proposition and trust is often reported as the *expected opinion* [21]. This translates the problem into degrees of belief, disbelief and uncertainty represented by $E^b_{ji} = b, E^d_{ji} = d, E^u_{ji} = u$ where $E^b_{ji} + E^d_{ji} + E^u_{ji} = 1$. Jøsang's belief model that utilizes Subjective Logic is popularly used to deal with such uncertainty in a proposition with binary state space, but having a multinomial evidence [21]. Josang's definition of trust as an opinion $\boldsymbol{\omega} = \{b, d, u, a\}$ is a quadruple where the components respectively correspond to the belief, disbelief, uncertainty, and relative atomicity such that $b, d, u, a \in [0, 1]$ and $b + d + u = 1$. The expected opinion pertinent to the positive interaction or belief is given as $E(\omega) = b + au$, where a is known as the relative atomicity which determines how uncertainty contributes to the final expected opinion. Without any information on the uncertainty dynamics in a system, the usual value of a is equal to inverse of the proposition state space cardinality, i.e., 0.5. Hence the expected opinion on the proposition that the node is cooperative or not is: $E^\omega_{ji} = E^b_{ji} + (a)E^u_{ji}$.

For the sake of illustrating the benefit of Subjective logic, the scenarios shown in Table III represent trust evidence on a particular time slot out of $N = 40$ channels for different nodes. Scenarios 1, 2, 6 and 7 have occurrences of mismatches while 3, 4 and 5 do not. Intuitively, we would expect 3, 4 and 5 to have higher trust than 1, 2, 6 and 7. However, scenario 4 has high number of uncertain ratings as opposed to 5. The previously proposed Optimistic Trust Model cannot capture

TABLE III
TRUST-OPINION TUPLE; N=40

Scenario	φ	β	μ	$E^{j,i}_{beta}$	E^{ω}_{ji}	w_{ji}
1	14	13	13	0.518	0.5116	0.045
2	19	18	3	0.513	0.5166	0.069
3	22	0	18	1.00	0.755	0.67
4	10	0	30	1.00	0.616	0.38
5	31	0	9	1.00	0.860	0.87
6	22	14	4	0.61	0.5929	0.31
7	14	22	4	0.38	0.398	−0.51

relative uncertainty in one value. Hence $E^{j,i}_{beta}$ from Eqn. (14) gives the same answer for scenarios 4 and 5. *This ambiguity is resolved by our Dirichlet expectation model*. The table values use $a = 0.5$ for demonstrating the dummy scenarios. In the results, we derive the appropriate value of a, my learning the uncertainty dynamics under various environmental paramters.

If we observe the corresponding values in the Conservative Trust Model, given by E^{ω}_{ji}, we observe that it captures the presence of high number of uncertain ratings by generating a trust value of 0.61 for scenario 4, whereas giving a higher value of 0.86 to scenario 5, thus effectively differentiating between scenario 4 and 5. We can also see that scenario 3 which has less uncertain ratings than 4 but more uncertain ratings than 5, has a trust value intermediate to the scenarios 4 and 5, thus preserving consistency in the rationale that given no evidence of mismatch, lower uncertain should be awarded with higher trust. Hence the value E^{ω}_{ji} does not have to necessarily depend on the assumption of uniformly random attacks or the non zero probability of not detecting a single channel's attack.

For the conservative model to work properly, we should expect that scenarios $1, 2, 6, 7$ have low trust values than $3, 4, 5$ as seen from Table III. However, among these scenarios where there is evidence of mismatches, scenario 6 has most number of matches and least undecided compared to the others. Hence scenario 6 achieves higher trust value than 1,2,7 but lower than scenario 4.

3) A Conservative Trust Metric: E^{ω}_{ji} is the expectation of the belief that the node j as seen by i, and is a number between 0 and 1. The system needs to perform a regression to determine, if node j is malicious or honest. We have used the generalized linear models (GLM) for this purpose. The expectation is a continuous variable, while the response/predictor variables is categorical (true/false, yes/no, etc.). In such cases, we need a link function to provide the relationship between the predictor variable (linear) and the mean of the distribution function defining the quality or regression score. This concept is well documented in bounded rationality, decision theory, prospect theory, and generalized linear classification where errors distribution is unknown. $r_{E^{\omega}_{ji}}$ is the linear predictor and E^{ω}_{ji} is the mean, the link between them is established by the following logic function. Hence, the use of Sigmoid log as a link function is justified. Without this, it will be impossible to guarantee a linearly separable trust distribution and a threshold based classification to segregate the malicious and honest nodes. A simple scaling function will not suffice. As regards to the final step which scales the value between -1 and +1, it is done to adhere to the standards of trust metric representation, which is represented as a real number either between 0 and 1 or between -1 and +1 as discussed in [19]. Hence we use a Sigmoid log function to map E^{ω}_{ji} on to a real line where non-trustworthy nodes have monotonically decreasing weights and trustworthy nodes have monotonically increasing weights. The log value based weight is given as:

$$r_{E^{\boldsymbol{\omega}}_{ji}} = log_2\left(\frac{E^{\boldsymbol{\omega}}_{ji}}{1 - E^{\boldsymbol{\omega}}_{ji}}\right) \tag{19}$$

We report the normalized conservative trust weight between $[-1, 1]$ using a scaling function that is given by:

$$w_{ji} = \begin{cases} 1 - e^{-|r_{E^{\boldsymbol{\omega}}_{ji}}|} & \text{if } r_{E_{ji}{}^{\boldsymbol{\omega}}} > 0; \\ -\left(1 - e^{-|r_{E^{\omega}_{ji}}|}\right) & \text{if } r_{E_{ji}{}^{\omega}} < 0; \\ 0 & \text{if } r_{E_{ji}{}^{\omega}} = 0 \end{cases} \tag{20}$$

where $w_{ji} \in [-1, 1]$.

VI. MALICIOUS NODE IDENTIFICATION

For malicious node identification, we use the absolute value of node j's final trust weights which is the average of all trust weights w_{ji} calculated by node j's neighbors. Hence long term average trust weights of a particular node j can be represented as w_j also known as *reputation* of a node, which is a collective measure of trustworthiness. The decision on a node being rendered as honest is usually done by a policy enforcement entity, who collects this weights from each node about its neighborhood and calculates the w_j. One such method of collection of trust ratings is the use of Distributed Hash Tables (DHT). In such a case reports of trust neighborhood, may be vulnerable to bad-mouthing attacks, but those issues have already been addressed in works like [12] and [22], hence we treat implementation issues pertinent to this as a black box. Those methods can be seamlessly integrated with our method.

A. Trust Update Over Time

For the given attack model (except for ON-OFF attacks), a cumulative equally weighted moving average for maintaining node reputation makes sense. This is because decision of isolation of a node needs to keep a long term history of behavior. As instantaneous trust value of node j as calculated by node its neighbor i at time t is $w_{ji}(t)$. The cumulative moving average is the average trust at time t for all of the interactions up to that point of time. Hence at any time t, a node's long term average trust, $w^{mavg}_{ji}(t)$ is updated as:

$$w^{mavg}_{ji}(t) = \frac{(t-1)w^{mavg}_{ji}(t-1) + w_{ji}(t)}{t} \tag{21}$$

The cumulative moving average is essential to characterize long term behavior or strategies of a node because it does not cause loss of information over time unlike exponential weighted moving average. The reputation of node j used to decide whether node j is malicious or not, is average of all $w^{mavg}_{ji}(t)$ pairs for each neighbor i who receives node j's spectrum data.

B. Trust Update for ON-OFF Attacks: Special Case

Till now we have discussed attackers who either use deterministic or probabilistic magnitude of attacks. In such attacks, there is no preference on which time slots the attacks will be launched. Thus for such cases, trust values over time can be updated as equally weighted moving average, that would reflect the true behavior over time. However, in ON-OFF attacks, nodes have preferences over time periods where a node may choose not to attack at all for some time and then attack for some time with a random magnitude. In such a case, both equally weighted moving average or exponentially weighted moving average would not reflect true behavior of the node. An equally weighted moving average will *lag* in reflecting such attacks, while weighted moving averages will enable a malicious node to *quickly recover or redeem its reputation*. In such cases, the trust management framework should be such that a node with a history of malicious or anomalous behavior should not be allowed to recover its trust value quickly even though it starts behaving well after a short burst of attack.

We propose a technique to deal with such ON-OFF attacks from a socially inspired concept that bad actions are remembered for longer than good actions. This forms the basis of our *asymmetric weighted moving average* scheme, where slots with instantaneous trust values $w_{ji}(t)$ lower than a threshold Γ_C are given more weight than slots where $w_{ji}(t)$ has higher values. The value of Γ_C is dictated by a system specific risk attitude and defines what can termed as sufficiently good behavior. For updating the trust values, there are two important aspects: the cumulative average, and the current trust value. Thus, we introduce four weighting factors χ_a, $\chi_{b_{max}}$, $\chi_{c_{min}}$ and χ_d such that $0 < \chi_a < 1$, $0 << \chi_{b_{max}} < 1$, $0 < \chi_{c_{min}} << 1$, and $0 < \chi_d < 1$. Note that the fact that $\chi_{c_{min}}$ is much much less than $\chi_{b_{max}}$ introduces an asymmetry. Now there may be four possible scenarios at time t with regards to ON-OFF attacks.
Case(a): $w_{ji}^{mavg}(t-1) > \Gamma_C$ and $w_{ji}(t) > \Gamma_C$
Case(b): $w_{ji}^{mavg}(t-1) > \Gamma_C$ and $w_{ji}(t) \leq \Gamma_C$
Case(c): $w_{ji}^{mavg}(t-1) \leq \Gamma_C$ and $w_{ji}(t) > \Gamma_C$
Case(d): $w_{ji}^{mavg}(t-1) \leq \Gamma_C$ and $w_{ji}(t) \leq \Gamma_C$

For Case (a), a cumulative average higher than Γ_C suggests a node is maintaining a sufficiently good behavior. If the current trust value is also higher than Γ_C then it suggests continuity of the good behavior. Hence continuing good behavior is rewarded with a high weighting factor χ_a to $w_{ji}(t)$ and low weightage given to $w_{ji}^{mavg}(t-1)$ using $1-\chi_a$ with χ_a being the *rewarding factor*. It helps a historically good node to improve or at least maintain its reputation if it behaved in a cooperative manner in this time slot t. Hence for Case (a) cumulative trust is updated as: $w_{ji}^{mavg}(t) = (1-\chi_a) \times w_{ji}^{mavg}(t-1) + \chi_a \times w_{ji}(t)$.

For Case (b), a cumulative average higher than Γ_C and $w_{ji}(t) \leq \Gamma_C$ suggests that a node maintained a sufficiently good behavior upto time $t-1$ and but has initiated some anomalous behavior in t. Hence all the good behavior until now needs to be forgotten and very high weight needs be given to current slot's anomalous behavior. Hence $w_{ji}(t)$ is weighted with a high value $\chi_{b_{max}}$ and $w_{ji}^{mavg(t-1)}$ is weighted using $1-\chi_{b_{max}}$ with $\chi_{b_{max}}$ being the *punishment factor*. The higher is the value of the punishment factor, quicker and more severe will be the system towards new evidences of malicious behavior. In such a case, the cumulative trust is updated as: $w_{ji}^{mavg}(t) = (1-\chi_{b_{max}}) \times w_{ji}^{mavg}(t-1) + \chi_{b_{max}} \times w_{ji}(t)$.

For Case (c), a cumulative average lower than Γ_C but a current trust value higher than Γ_C signify a node whose current behavior is cooperative but has a history of anomalous behavior. Hence we assign $w_{ji}(t)$ a very low weight $\chi_{c_{min}}$ and assign $w_{ji}^{mavg(t-1)}$ a wight of $1-\chi_{c_{min}}$ with $\chi_{c_{min}}$ being the *redemption factor*. It controls how fast or slow a node with malicious history can redeem itself by demonstrating good behavior for a sufficiently long time. In such a case, the cumulative trust is updated as: $w_{ji}^{mavg}(t) = (1-\chi_{c_{min}}) \times w_{ji}^{mavg}(t-1) + \chi_{c_{min}} \times w_{ji}(t)$.

For Case (d), both cumulative average and current trust value of node j are below Γ_C indicating continuing anomalous behavior. In such a case, we use a weighting factor of χ_d to $w_{ji}(t)$ and $1-\chi_d$ to $w_{ji}^{mavg(t-1)}$, with χ_d being the *retrogression factor*. In such a case, the cumulative trust is updated as: $w_{ji}^{mavg}(t) = (1-\chi_d) \times w_{ji}^{mavg}(t-1) + \chi_d \times w_{ji}(t)$.

The above scheme termed as the *asymmetric weighted moving average* is effective in defending against ON-OFF attacks which is not possible using equally weighted or exponential weighted moving averages.

C. Machine Learning Based Classification Threshold Design

A classification threshold needs to be computed for trust based identification of malicious nodes. We propose the threshold design using a supervised machine learning approach that learns all network, radio, and topological parameters that affect the trust value distribution. The threshold learning has a training phase and a model selection phase. The purpose of a training phase is to learn/predict different candidate thresholds for different training sets. A training phase is like a controlled environment where the defender performs some small scale experiments with a set of *training nodes* by varying few network parameters. Few of such training nodes are programmed to act as malicious while the others behave as honest. The proposed model is then applied, and the nodes programmed as honest and malicious nodes end up with some trust values. Subsequently, both the trust value and label of each node (ground truth in training) is supplied to the Support Vector Machine (SVM) classifier. The SVM classifier learns the difference between malicious and honest nodes in terms of their trust values. Based on this learning, the SVM outputs an optimal hyperplane guarentees maximum separation between the honest and malicious labels. This hyperplane produces a threshold for each training set. The most appropriate training set is one that exhibits least under or over fitting is selected. The corresponding threshold of the *selected training set* is then applied for classification in an unknown real deployment known as testing set as shown in the results.

We generate training data sets for different path loss environments ω and varying P_{attack} with worst case standard deviation of shadow fading. Our objective is to find an optimal threshold Γ_C that can decide whether a node is malicious or not. To design appropriate training sets, we need to explore the effects of these features upon the trust values.

TABLE IV
EFFECT OF PATHLOSS ON UNCERTAINTY

Pathloss	Average E_μ
3	0.166065
4	0.426087
5	0.305750
5.5	0.200618

Effects of pathloss environment: The variation of the pathloss exponent affects the degree of uncertainty in the output of the anomaly monitoring technique. Our experiments show that a network with pathloss exponent of 4 (neither high or low) induces maximum uncertainty, while pathloss exponents lower or higher than 4 (say 3 or 5), lowers the average degree of undecided. The reason for this is that if signals decay too fast or too slow for higher and lower pathloss exponents, the chances of both P_{high} and P_{low} being either above or below the particular normalizing threshold γ_{th} within the radius s_{ij} is increased. Given all the other factors remain the same, and if the pathloss exponent is neither too high or too low, the chances of both P_{high} and P_{low} being above or below γ_{th} *decreases*. This is evident from our experiments which calculates the average degree of uncertainty for all nodes across the network for different pathloss environments listed in Table IV. While pathloss exponent as low as 3 or as high as 5.5, generates lower average E_μ of 0.16 and 0.20 respectively, an intermediate pathloss exponent of 4, produces E_μ as high as 0.42. Lack of information increases the difficulty of a classification problem. Hence, classification becomes harder when pathloss is around 4 and easier when the pathloss exponents are on the extremes. Thus, we are motivated to use training sets *considering different pathloss environments*.

Effects of magnitude of attack: It is intuitive that under an effective monitoring mechanism, the more a node attacks the more it exposes itself for detection. Though a malicious node can decrease its magnitude of attack to evade detection, it beats the purpose of attacking the network. Thus, the malicious nodes will have to strike a balance between attacking and avoiding detection. The optimal attack strategy in multi-channel systems is 0.5 as shown in [24]. In general, if we can detect for lower magnitudes of attack, we can detect for higher magnitude of attack as well. Hence, we use training data sets mostly considering lower magnitudes of attack to thwart sub optimal conservative attackers. Hence, we choose magnitude of attack as 0.3. This value is much less than the half way value of 0.5. Hence, it induces less underfitting. However very low magnitude for training sets will over fit and we may classify honest nodes as malicious because some percentage of channels for honest nodes are altered due to P_f, P_m, and f_l.

Training data sets: We use one training set for pathloss $\omega = 3$, 4 and 5 each with magnitude of attack 0.3. We assume the worst case standard deviations of $\sigma_s = 6$ dB and $\sigma_l = 12$ dB due to shadow fading. We observe the trust values of the honest and malicious nodes. We run a support vector machine (SVM) over training examples which maps the trust values into support vectors and find the optimal hyper-plane which in our case is a single line due to the linear nature of the data with only one feature, i.e., the trust value. Figures 2, 3(a), 3(b),

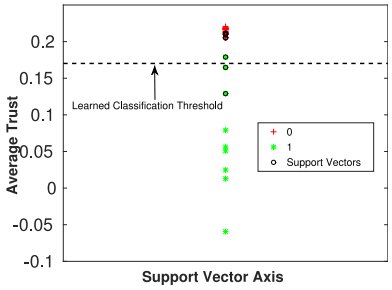


Fig. 2. Training Phase Threshold Prediction: Pathloss=4; $P_{attack} = 0.30$.

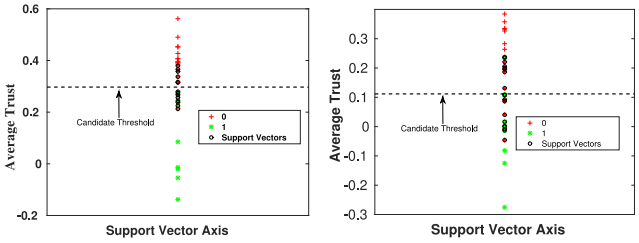


Fig. 3. Alternative Candidate Thresholds (a) Pathloss=3, $P_{attack} = 0.3$ (b) (a) Pathloss=5, $P_{attack} = 0.3$.

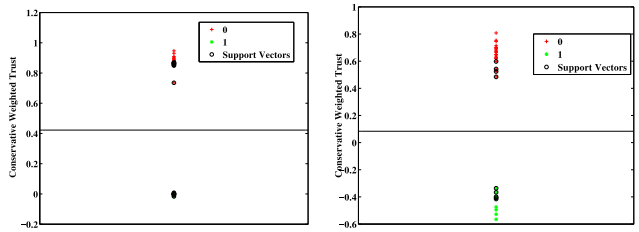
show results for the candidate thresholds obtained by a Support Vector Machine for each training data set. $+$ represents the labels corresponding honest nodes and $*$ represents labels corresponding to the malicious node. The solid line separating is the output threshold learned by the Support Vector Machines based on the labels in the training set. The threshold predicted in this way, will be applied to a testing set of different network features and unknown labels of nodes.

Rationale for model selection: The lower region of the SVM contains labels corresponding to malicious nodes and the upper region contains labels that correspond to the honest nodes. Our objective is to mimic some worse case scenarios for classification. Thus we emphasize on the lower probabilities of attack 0.3 where classification is harder and an intermediate path loss environment with highest inherent uncertainty in the evidence. Hence, SVM output of Figure 2, is chosen as our classification threshold $\Gamma_C = 0.17$.

Alternative candidate thresholds are shown in Fig. 3(a), and Fig. 3(b). To prove why the above training sets with $P_{attack} = 0.30$ are sufficient, we plotted Fig. 4(a) and Fig. 4(b), where $P_{attack} = 0.50$ and 0.80 respectively. In both cases, their thresholds are much lesser and the difference between support vectors of honest and malicious labels are higher. This is much easier to classify, hence this option does not dominate the threshold for lower magnitudes of attack.

VII. TRUST BASED ROBUST FUSION

Though steady state values of trust are ideal for node identification and classification, such values do not necessarily contribute towards robust fusion given our adversarial model. In dynamic systems, *waiting for convergence* for filtering out spurious reports is not an option. This is also relevant in an ad-hoc CR network in three ways. First, nodes may be highly



(a) (b)

Fig. 4. Under Fit Training Sets (a) Pathloss=3, $P_{attack} = 0.50$ (b) Pathloss=3, $P_{attack} = 0.80$.

mobile, hence the neighbors of a node are not long lasting entities. At the same time however, a decision on the channel occupancy needs to be made from the reports of current neighbors. Hence maintaining history of updates of trust value of neighbors may not be a prudent idea in a distributed CR network. Second, fusion performance is dependent on scenario in the current time slot. The past has no bearing on the spectrum occupancy of the present slot or prior nature of honesty/cooperative behavior as far fusion is concerned. Third, the proposed attack measure P_{attack} is a long term value, but a particular realization of P_{attack} at a particular time, may be different from the mean value of P_{attack}. For example, a malicious node has $P_{attack} = 0.60$, but at a particular time slot only 0.3 fraction of channel may have been attacked. In that case, at this time slot it has contributed on 70% of the channels. In such a case, if we isolate this report based on long term reputation based exclusion, we will lose the majority of honest opinions along with the minority falsified opinions. However, instantaneous or transient trust or reputation is an index of honesty/cooperative behavior on the current interaction. Since only current interactions are important as far as spectrum sensing usage reports are concerned, steady state trust values should not be used as a metric for exclusion of spectrum sensing reports on each time slot.

Using the computed *instantaneous trust* coefficients, we study the performance of two fusion schemes: Trust based fusion and Conservative trust based fusion. We compare their performance benefit by comparing it with Blind Fusion.

A. Blind Majority Voting Based Fusion

For blind fusion, node i considers all its neighbors to be honest and includes B^j_{adv} from all its neighbors along with its own B^i_{act}. We formally define Blind Fusion as $BF^i_{blind} = \nabla[B^j_{adv} \oplus B^i_{act}]$, $j \in N_i$ where ∇ is the operator for majority voting rule. Majority voting is a popular fusion rule where final fused inference on a channel is based on what at least half the neighboring nodes advertise with all the nodes treated equally. $\oplus$ is the operator for combination.

B. Optimistic Trust Based Fusion

We propose a fusion scheme whereby we only consider neighboring nodes whose $E^{j,i}$ is higher than some trust threshold, Γ_{opt}. (Later in Section VIII, we show how to find the optimal threshold). Thus, for trust-based fusion, node i only considers those neighbors whose $E^{j,i} \geq \Gamma_{opt}$. In effect, the fusion is done with information from trusted nodes only.

$$\text{If} \quad E^{j,i}_t \begin{cases} \geq \Gamma_{opt} & \text{Node } j's \text{ report trusted;} \\ < \Gamma_{opt} & \text{Node } j's \text{ report not trusted.} \end{cases} \tag{22}$$

C. Conservative Trust Based Fusion

Similar, to the above, we propose to only consider nodes in the conservative model, that are above a threshold Γ^c_{opt} such that

$$\text{If} \quad w^t_{ji} \begin{cases} \geq \Gamma^c_{opt} & \text{Node } j's \text{ report trusted;} \\ < \Gamma^c_{opt} & \text{Node } j's \text{ report not trusted.} \end{cases} \tag{23}$$

D. Performance Analysis Measures

We evaluate the performance of robust fusion and malicious node detection in terms of the following measures.

Percentage of mismatches: We define Trust Based Fusion result as: $TBF^i = \nabla[TFS_i \oplus B^i_{act}]$; where TFS_i is the trusted fusion set of binary vectors accumulated by node i using Eqn. (22), which includes B^j_{adv} of trusted nodes only.

Although the nodes are not aware of the ideal scenario, we are aware of what would have been the ideal fusion result, which is the case when for all node $j \in N_i$, $B^j_{act} = B^j_{adv}$, so we define Ideal Fusion result for node i, $BF^i_{ideal} = \nabla[B^j_{act} \oplus B^i_{act}]$. This is later used for comparing the performance of fusion with fusion schemes 1 and 2 by measuring deviation from ideal result.

Percentage of true negatives and accurate detection: These measures are used to establish how well our malicious node identification works compared to existing research. Percentage of True Negative is the number of malicious nodes successfully captured from all malicious nodes. A more strict measure is the percentage of accurate detection that considers the possibility of honest nodes being declared as malicious.

VIII. Simulation Results

To validate the trust model, we conduct extensive simulation experiments. We consider a primary network of 600×600km as shown in Fig. 13. 40 primary transmitter towers are poisson distributed near the central areas to avoid edge effects. The smaller secondary networks are contained within the primary network as sub-networks. This ensures a good mix of available and non-available channels. For the training set, we consider a 60×60 km grid with 30 randomly scattered (Poisson) nodes– 9 of which are programmed to be malicious. Both I_{attack} and P_{attack} described in Section III are varied from 0.10 to 0.95. The malicious nodes are non-collaborative by default on the channels they falsify unless explicitly mentioned otherwise and the number is dictated by I_{attack} and P_{attack}. We also provide some results from malicious adversaries perspective. Each node scans 40 channels and has a report sharing radius of 20 km units. The secondary nodes are considered stationary.

For the secondary network corresponding to the testing set, we consider a sub-region of 200×200 km with 100 nodes, within the primary network. The malicious nodes in the testing set are divided into three groups, each with low, medium and high magnitudes of attacks. The testing set contains 30% malicious nodes. We also vary the path loss exponent ω from 3

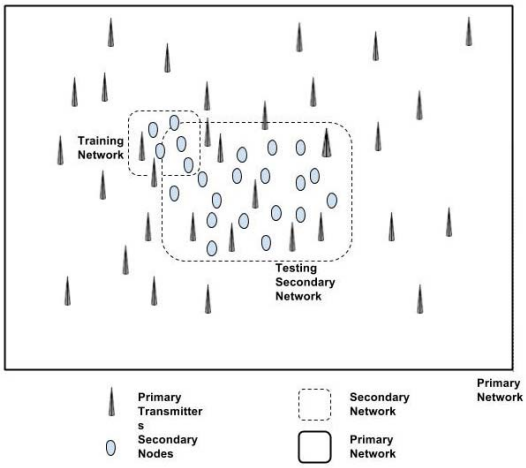


Fig. 5. Simulation Scenario showing primary and secondary networks.

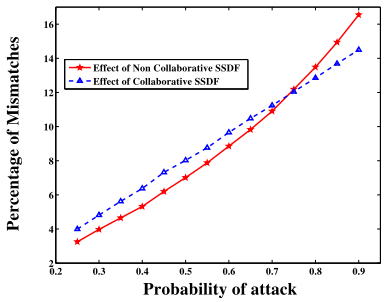


Fig. 6. Effects between collaborative and non-collaborative SSDF.

to 5, while the fading standard deviations are 5 dB and 10 dB respectively. We also provide some results from malicious adversaries perspective. Sections VIII-A have the details of attack emulation for collaborative vs. non-collaborative SSDF attacks and P_{attack} vs. I_{attack} respectively. Section VIII-C shows how results are affected with varying pathloss environments and different emulated magnitudes of attack.

A. Non-Collaborative vs. Collaborative SSDF Attacks

In Fig. 6, we compare the damages inflicted by collaborative SSDF versus non-collaborative SSDF on the network in terms of the percentage of mismatches for blind fusion without any defense. We observe that collaborative attack is able to damage more for most P_{attack} values, except when $P_{attack} > 0.80$. From Fig. 6, the conclusion is that for P_{attack}, collaborative SSDF is a better attack strategy in terms of the deviations it causes from the ideal result. However, collaborative SSDF becomes less effective than non-collaborative counterpart when $P_{attack} > 0.8$. This is because when P_{attack} is high for all malicious nodes, there will automatically be many common channels in the attacked set, even when attacked independently. Hence an implicit collaboration follows. However, it must be noted that cost of collaboration between malicious nodes is higher than non-collaboration and may not always be feasible. Magnitude for collaborative attacks is always dictated by I_{attack} while for non-collaborative it can be either I_{attack} or

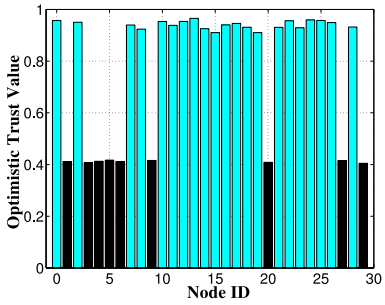


Fig. 7. Trust of honest and dishonest nodes for $P_{attack} = 0.60$.

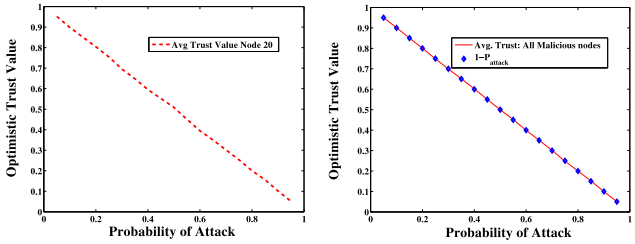


Fig. 8. (a) Node 20's (malicious node) trust (b) All malicious nodes trust.

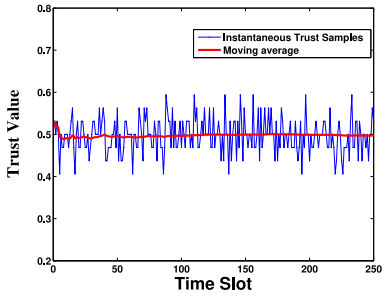


Fig. 9. Instantaneous and average trust for $I_{attack} = 0.50$: Node 20 as observed by Node 10.

P_{attack}. The comparison of results between I_{attack} and P_{attack} is given later in Figures 9 and 10.

B. Optimistic Trust Model

In this subsection, we discuss all relevant results for the optimistic trust model.

Trust measurement: In Fig. 7, we observe the difference in trust distribution between malicious and honest nodes when P_{attack} is 0.6. The trust is evaluated as the average by all its neighboring nodes after 500 time slots. It is evident that malicious nodes have trust values significantly lower than those of the malicious nodes.

In Fig. 8(a), we show how the trust varies for possible values of P_{attack} for a *particular* node (node no. 20 in this case). We also show the average trust for all nodes in Fig. 8(b). As expected, higher attack probabilities result in low trust for both cases. An interesting and intuitive observation from Figs. 7, 8(a) and 8(b) is that the trust of malicious nodes converge to $1 - P_{attack}$.

Comparison between P_{attack} and I_{attack}: For I_{attack}, the total number of channels remain same in every time slot although

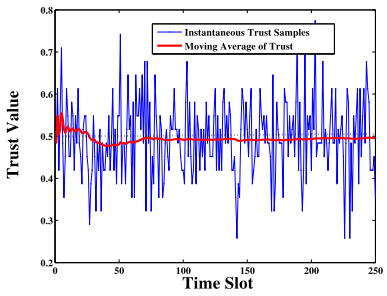


Fig. 10. Instantaneous and average trust for $P_{attack} = 0.50$ for node 20 as observed by node 10.

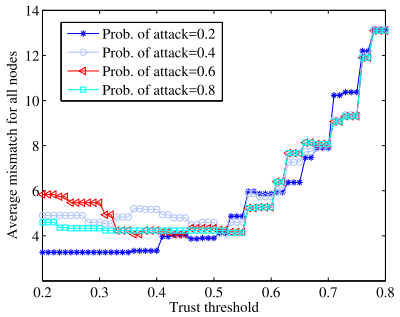


Fig. 11. Optimal threshold (Γ_{opt}) for trust based fusion.

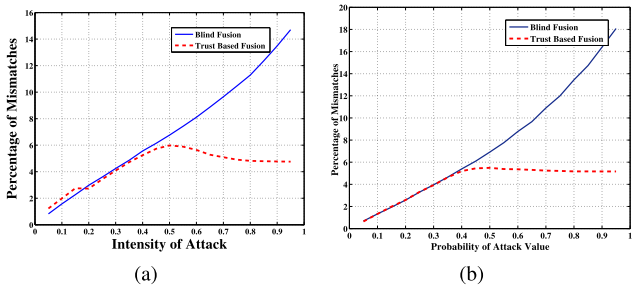


Fig. 12. Blind Fusion vs Trust Based Fusion (a) Under I_{attack} (b) Under P_{attack}.

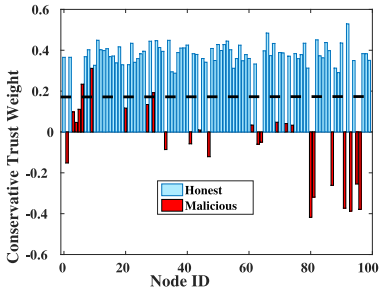


Fig. 13. Testing Set Worst Case Performance: Pathloss=4.2.

individual channels attacked vary. On the other hand, P_{attack} induces more uncertainty where the number of channels as well as channels attack vary. In Figures 9 and 10, we see how the instantaneous and average trust varies for I_{attack} and P_{attack}, respectively. The higher variance, and therefore slower convergence (80 time slots instead of 40 for I_{attack}), for P_{attack} makes it difficult for defenders to compute the trust values quickly. Thus, from a malicious node's perspective, it is advantageous to employ P_{attack}.

Choosing the optimal threshold: For the optimistic system, the value of threshold Γ_{opt} for trust based fusion using different candidate thresholds (Γ) ranging from 0.2 to 0.8.

Fig. 11, shows that for very low values of hypothetical thresholds, there are more mismatches since most of the malicious nodes are included for fusion. However, as we increase this threshold, malicious nodes start getting discarded and mismatches decreases. However, when the threshold is very high (above 0.6), the mismatches increase again because information from nodes with higher trust values also get discarded. The minimum number of mismatches occurs for the range of threshold values from 0.45 to 0.51. For the rest of the results, we use $\Gamma_{opt} = 0.5$ since beyond this point the damage is more than the benefits of cooperation.

Blind fusion vs trust based fusion: In Figures 12(a) and 12(b), we show the percentage of mismatches for all P_{attack} values ranging from as low as 0.05 till 0.95. We observe that the percentage of mismatches are far less for trust based fusion which filters out spectrum reports from potentially dishonest nodes rather than blind fusion. More specifically, we see that the percentage of mismatches is always less than 3% of the total channels for both I_{attack} and P_{attack} equal to 0.2. For $P_{attack} = 0.05$, the percentage of mismatches is 1% for I_{attack} and 0.7% for P_{attack}. For $P_{attack} = 0.10$, it is about 1.2%. Hence, it is clear that for lower magnitudes of attack the network is not significantly damaged hence they are allowed, and hence the blind fusion and trust based fusion have almost similar mismatches. However, when the attack magnitude increases, the trust values of malicious nodes fall below the desired threshold Γ_{opt} and their false opinions get filtered out decreasing mismatches.

C. Conservative Trust Model: Malicious Node Identification

We consider a network with 100 nodes a fraction of which is malicious. The malicious are divided into three groups with P_{attack} 0.3, 0.5 and 0.8 respectively. The threshold selected from the chosen model should be able to capture such nodes, under any pathloss exponent. This testing set had 27 randomly chosen malicious nodes and each group has 9 nodes. We consider pathloss exponent of 4.2, 3.5 and 5.2.

Identification of malicious nodes (Pathloss=4.2): We use $\Gamma_C = 0.17$, for the testing set with pathloss 4.2. This was inferred from training phase threshold selection. We observe that 27 nodes regardless of their P_{attack} have been accurately classified as malicious as they all have trust values below 0.17 as shown in Fig. 13. Two malicious nodes narrowly miss detection in mainly due to low P_{attack}.

Identification of malicious nodes (Pathloss=3.5 and 5.2): Now we show the performance for other environments with $\omega = 3.5$ and $\omega = 5.2$. with the inferred $\Gamma_C = 0.17$. Fig. 14(a) and Fig. 14(b), show that in the worst case fading, the classification is accurate, and is able to distinguish most of the honest nodes from malicious ones.

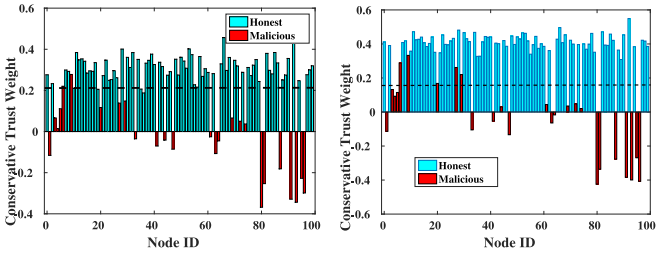


Fig. 14. Testing Set Worst Case Performance (a) Pathloss=3.5 (b) Pathloss=5.2.

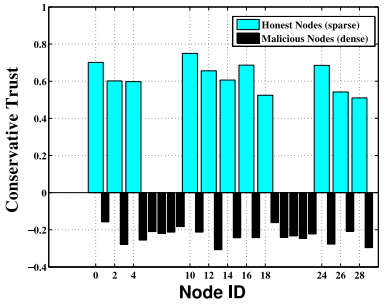


Fig. 15. Performance: High density ($\rho_{mal} = 60\%$) of collaborative malicious nodes with $P_{attack} = 0.80$.

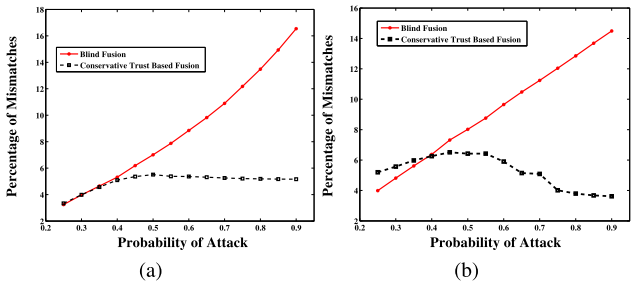


Fig. 16. Conservative trust based fusion (a) Non-collaborative P_{attack} (b) Collaborative P_{attack}.

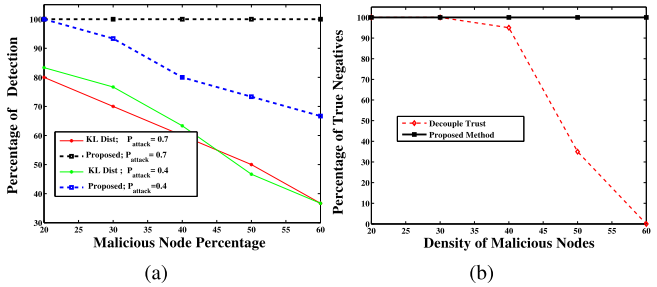


Fig. 17. Comparison of Proposed Trust Model (a) With KL distance method (b) With majority voting based exclusion.

Higher densities of collaborative SSDF: Normally, Kullback Leibler divergence, voting based reputation, entropy and average SNR divergence techniques do not work when fraction of collaboratively malicious to the total participating nodes are above 50%. In distributed networks, local topology variations may cause a node to have more malicious neighbors than honest. To test whether our proposed model works in such situations, we simulate with minority honest nodes (11 of them) and malicious nodes (19 of them). From Fig. 15, we can see that there is a significant difference between honest and malicious even under high density (60%) of collaborative malicious nodes, an improvement from voting based exclusion.

D. Robust Fusion Using Conservative Trust Weights

Fig. 16(a) shows the performance of the proposed conservative trust based fusion model as opposed to blind fusion. We use the threshold of 0.50 with the rationale that nodes that damage more than when they cooperate is not considered. For conservative fusion, we choose 0, which is the equivalent of 0.50 for the conservative trust model when trust values are bounded between the interval [-1,1], because $ln(\frac{0.5}{1-0.5}) = 0$. As expected, trust based fusion has lower number of mismatches. The same also holds true even if the malicious nodes launch collaborative attacks where they agree upon select channels. However, when P_{attack} is employed, the trust based fusion gives more mismatches than blind fusion as can be observed in Fig. 16(b). Hence an important inference is that when P_{attack} is low (< 0.5) for collaborative SSDF, trust based fusion that disregard nodes based on their trust values in not an effective approach. This is effective for selfish SSDF attacks, where the magnitude of attack is low and nodes collaborate to falsify on specific channels. Thus, a different defense strategy is required for very low intensity collaborative SSDF attacks.

E. Comparison With Existing Research

We seek to compare the benefits of our proposed trust based malicious node detection scheme with a few existing research works. We use our conservative trust model with KL divergence [25] or majority voting based exclusion (decouple) method [11]. We compare the percentage of accurate detection over various fraction of malicious nodes for different P_{attack}. In Fig. 17(a), we observe that under various values of fraction of malicious nodes and P_{attack}, our method using w_j value yields much better results than existing researches discussed in [24] and [25] particularly under high fractions of malicious nodes or high P_{attack}. In Fig. 17(b), we compare our work with another recent work [11]. We report significantly high true negative detection percentage across different malicious node fractions.

F. Defending Against ON-OFF Attacks

For ON-OFF attacks, we limit our simulation study to the trust dynamics of a particular node 20 which launches ON-OFF attacks in five stages over 500 slots. In 'Stage 1', it behaves cooperatively and does not attack on any time slots from $t = 0$ to $t = 100$. In 'Stage 2' the node attacks with a random fraction of channels on each time slot from $t = 101$ to $t =$ 150-th slot. In 'Stage 3' it does not attack for the next 100 slots till $t = 250$. In 'Stage 4', it attacks from $t = 251$ to $t = 300$ just like Stage 2. In 'Stage 5', the node does not attack for the next 200 slots till $t = 500$. We

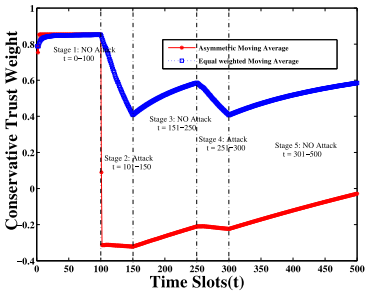


Fig. 18. Asymmetric moving average vs equal weighted moving average.

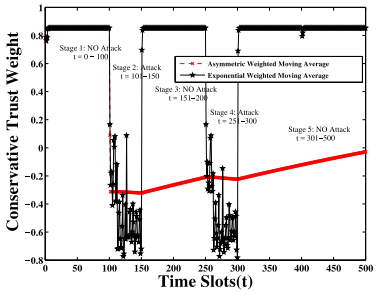


Fig. 19. Asymmetric moving average vs. exponentially weighted moving average.

plot the results of ON-OFF attacks seen by one of its neighbor, node 29 using equations from the asymmetric weighted moving average discussed in Section VI-B. We compare the results with other popular trust update schemes and justify suitability of asymmetric averaging with regard to ON-OFF attacks.

Choice of weighing factors and threshold: The weighing factors χ_a, $\chi_{b_{max}}$, $\chi_{c_{min}}$, and χ_d are chosen as 0.999, 0.999, 0.001 and 0.001. We can verify that this satisfies the conditions: $0 < \chi_{c_{min}} << \chi_{b_{max}} < 1$, $0 < \chi_a < 1$, and $0 < \chi_d < 1$. From the skewed values of the weighing factors $\chi_{c_{min}}$ and $\chi_{b_{max}}$, it justifies the asymmetry that we provide by giving negative behaviors a very high weightage and positive behavior and very low weightage on the first occurrence of negative behavior. The choice can χ_a and χ_d can be used to control the rate of trust redemption. If a system requires slower trust redemption that lower value of χ_a and lower value of χ_d is necessary. Since there is no particular magnitude of attack we keep the mid point between the trust value range $(-1, +1)$ as $\Gamma_C = 0$. However, Γ_C can be adjusted according to the requirements of the system. More conservative systems will have $\Gamma_C > 0$. Different values of χ_{min} and χ_{max} can be chosen to ensure more fairness to nodes in a network inherently susceptible to more bit flips due to noise.

Comparison with equal weighted moving average: In Fig. 18, we show how the proposed asymmetric weighted moving average performs as opposed to the equal weighted moving average. We observe that at Stage 1 with no attacks, both schemes preserve a high trust value, but when attacks start from the 101st time slot for the next 50 slots, asymmetric weighted moving average ensures cumulative trust is decreased more rapidly and preserves a low value. Equal weighted moving average is slow to react due to the node having behaved well in the first 100 slots. This happens because once current value in a slot is less than zero, the model forgets previous high reputation through a very low value $1 - \chi_{b_{max}} = 0.001$ and expresses extremely high weight $\chi_{b_{max}} = 0.999$ to the current values from the 101st time slot, thus causing the cumulative trust at stage 2 to decrease rapidly. In the beginning of Stage 3, when the attack ceases, we see that trust value reflected by asymmetric average is low enough (-0.25) to reflect node's malicious history while equal weighted moving average fails to capture because the ON-OFF attack ratio is 1:2, i.e., more slots with no attacks. This happens because, previous cumulative trust of less of than zero at the end of Stage 2 is given a very high weight compared to current honest behavior. It prevents the trust values to improve even during honest behavior.

In Stage 4, when attack starts after honest behavior for 100 slots, we see the significant difference between the trust values of the two schemes is preserved. Same is the case in Stage 5, where the reasons of a very slow increase in trust values under asymmetric average compared to equal weighted average is the difference in weighing factors assigned to previous and current trust values. Hence, we conclude that asymmetric weights can offer the benefits not provided by equal weighted moving average in terms of reacting quickly to ON-OFF attacks and preserving a low trust value of a malicious node. Through this scheme we have ensured that even though it targets only 100 out of 500 slots, the model can identify such nodes.

Comparison with exponential weighted moving average: The major criticism of exponentially weighted moving average was that although it reacts quickly when attacks start, it also forgets malicious behavior as quickly as it reacts. This is inappropriate because a malicious node should not be allowed to increase its trust value quickly unless it engages in a long period of honest behavior to redeem its trust. The key point where a difference is created is case(c) of the ON-OFF defense schema where we provide very low value to honest behavior after a period of dishonest behavior. Hence it's cumulative trust value hardly increases. In Fig. 19, we do not see much difference in Stage 1 due to no attacks. Also there is not much difference is Stage 2 as there more weight given to new trust values by both models. However, in Stage 3, exponential weighted moving average allows the malicious node to quickly recover its trust value owing to forgetting old values. On the other hand, asymmetric average selectively does not forget old values that are low. This happens because, previous cumulative trust of less of than zero (selected Γ_{on_off}) at the end of Stage 2 is given a very high weight compared to current honest behavior. It prevents the trust values to improve even in the period of honest behavior. We see that for all subsequent stages the exponentially weighted averages oscillates between high and low values, but asymmetric average preserves a low value all the while at the same time allowing fairness by allowing very slow increase of cumulative trust at stage 5 owing to its continuous good behavior for 200 slots. This provision also helps nodes which experience noise to eventually redeem their trust on experiencing good transmission channels as we see next.

IX. Conclusion

In this paper, we proposed a spatio-spectral anomaly detection technique that is able to gather evidence that reflects malicious behavior of nodes in a distributed cognitive radio network without location information. Based on the evidence gathered from the anomaly detection technique, we propose two trust models: an optimistic one and a conservative one. We show that the optimistic trust heuristic can be approximated by a modified coarsened beta distribution. Subsequently, we use a Dirichlet distribution inspired trust model that is conservative in its assumptions. We propose a learning approach towards identification of malicious nodes under different pathloss environments and magnitudes of attack. Results exhibit that the proposed models perform significantly better than other models under a variety of pathloss environments, different densities of malicious nodes, varied magnitudes of attacker collaboration, and attack models such as, probabilistic SSDF, deterministic SSDF, and ON-OFF attacks. We also show significant improvement in trust based fusion where we disregard possible reports from potentially less trustworthy nodes using instantaneous trust values for both trust models. We also analyze different attack measures like P_{attack} and I_{attack} and discuss which is better technique from the malicious user's perspective. We also show the effects of collaboration, and non-collaboration from malicious nodes in terms of the damages they can cause to the network. As part of future work, we will perform optimization and design trade-offs based on a detailed cost-benefit analysis of the ensuing complexity and overhead of our proposed distributed trust management scheme on distributed DSA networks.

References

[1] Accessed on Dec. 2016. [Online]. Available: http://people.seas.harvard.edu/~jones/es151/prop_models/propagation.html

[2] (May 2008). *CISCO Wi-Fi Location-Based Services 4.1 Design Guide*. Accessed on Dec. 2016. [Online]. Available: http://www.cisco.com/en/US/docs/solutions/Enterprise/Mobility/WiFiLBS-DG.html

[3] H. L. Bertoni, *Radio Propagation for Modern Wireless Systems*. Upper Saddle River, NJ, USA: Prentice-Hall, 2000.

[4] S. Bhattacharjee, S. Sengupta, and M.Chatterjee, "Vulnerabilities in cognitive radio networks: A survey," *J. Comput. Commun.*, vol. 36, no. 13, pp. 1387–1398, Jul. 2013.

[5] S. Bhattarai *et al.*, "Defining incumbent protection zones on the fly: Dynamic boundaries for spectrum sharing," in *Proc. IEEE DySPAN*, Stockholm, Sweden, 2015, pp. 251–262.

[6] S. Bhattarai, J.-M. J. Park, B. Gao, K. Bian, and W. Lehr, "An overview of dynamic spectrum sharing: Ongoing initiatives, challenges, and a roadmap for future research," *IEEE Trans. Cogn. Commun. Netw.*, vol. 2, no. 2, pp. 110–128, Jun. 2016.

[7] S. Bhattacharjee, S. Debroy, and M. Chatterjee, "Trust computation through anomaly monitoring in distributed cognitive radio networks," in *Proc. IEEE PIMRC*, Toronto, ON, Canada, Sep. 2011, pp. 593–597.

[8] S. Bhattacharjee and D. C. Marinescu, "A cloud service for trust management in cognitive radio networks," *Int. J. Cloud Comput.*, vol. 3, no. 4, pp. 326–353, Jan. 2014.

[9] S. Bhattacharjee, M. Chatterjee, K. Kwiat, and C. Kamhoua, "Bayesian inference based decision reliability under imperfect monitoring," in *Proc. IFIP/IEEE Int. Symp. Integr. Netw. Manag. (IM)*, Ottawa, ON, Canada, 2015, pp. 1333–1338.

[10] D. Cabric, S. M. Mishra, and R. W. Brodersen, "Implementation issues in spectrum sensing for cognitive radios," in *Proc. Asilomar Conf. Signals Syst. Comput.*, vol. 1. Pacific Grove, CA, USA, 2004, pp. 772–776.

[11] Y. Cai *et al.*, "Decoupling trust and wireless channel induced effects on collaborative sensing attacks," in *Proc. IEEE DySPAN*, McLean, VA, USA, Apr. 2014, pp. 224–235.

[12] A. B. Can and B. Bhargava, "SORT: A self-organizing trust model for peer-to-peer systems," *IEEE Trans. Depend. Secure Comput.*, vol. 10, no. 1, pp. 14–27, Jan./Feb. 2013.

[13] R. Chen, J.-M. Park, and K. Bian, "Robust distributed spectrum sensing in cognitive radio networks," in *Proc. IEEE INFOCOM*, 2008, pp. 1876–1884.

[14] K. Cho, B.-G. Lee, and D. H. Lee, "Low-priced and energy-efficient detection of replicas for wireless sensor networks," *IEEE Trans. Depend. Secure Comput.*, vol. 11, no. 5, pp. 454–466, Sep./Oct. 2014.

[15] D. Gambetta, "Can we trust trust?" in *Trust: Making and Breaking Cooperative Relations*. Oxford, U.K.: Univ. Oxford, 2000, ch. 13, pp. 213–237.

[16] A. Ghasemi and E. S. Sousa, "Collaborative spectrum sensing for opportunistic access in fading environments," in *Proc. IEEE DySPAN*, Nov. 2005, pp. 131–136.

[17] A. Jøsang and J. Haller, "Beta reputation systems," in *Proc. 15th Bled E-Commerce Conf.*, Apr. 2002, pp. 1–14.

[18] A. Josang and J. Haller, "Dirichlet reputation systems," in *Proc. 2nd Int. Conf. Availability Rel. Security (AREAS)*, Vienna, Austria, Apr. 2007, pp. 112–119.

[19] A. Jøsang, R. Ismail, and C. Boyd, "A survey of trust and reputation systems for online service provision," *J. Decis. Support Syst.*, vol. 43, no. 2, pp. 618–644, 2007.

[20] A. Jøsang and Z. Elouedi, "Interpreting belief functions as Dirichlet distributions," in *Proc. Eur. Conf. Symb. Quant. Approaches Reason. Uncertainty*, Hammamet, Tunisia, Nov. 2007, pp. 393–404.

[21] A. Jøsang, "A logic for uncertain probabilities," *Int. J. Logic Fuzziness Knowl. Based Syst.*, vol. 9, no. 3, pp. 279–311, Jun. 2001.

[22] M. Misaghi, E. Da Silva, and L. C. P. Albini, "Distributed self-organized trust management for mobile ad hoc networks," in *Communications in Computer and Information Science*, vol. 293. Berlin, Germany: Springer, 2012, pp. 506–518.

[23] K. Pahlavan and P. Krishnamurthy, *Principles of Wireless Networks: A Unified Approach*. Upper Saddle River, NJ, USA: Prentice-Hall, 2002.

[24] A. S. Rawat, P. Anand, C. Hao, and P. K. Varshney, "Countering Byzantine attacks in cognitive radio networks," in *Proc. IEEE ICASSP*, Dallas, TX, USA, Mar. 2010, pp. 3098–3101.

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

[26] Y. Sun, Z. Han, and K. Liu, "Defense of trust management vulnerabilities in distributed networks," *IEEE Commun. Mag.*, vol. 46, no. 2, pp. 112–119, Feb. 2008.

[27] W. Wang, H. Li, Y. Sun, and Z. Han, "CatchIt: Detect malicious nodes in collaborative spectrum sensing," in *Proc. IEEE GLOBECOM*, Honolulu, HI, USA, 2009, pp. 1–6.

[28] F. R. Yu, H. Tang, M. Huang, Z. Li, and P. C. Mason, "Defense against spectrum sensing data falsification attacks in mobile ad hoc networks with cognitive radios," in *Proc. IEEE MILCOM*, Boston, MA, USA, 2009, pp. 1–7.

Shameek Bhattacharjee received the B.Tech. degree in information technology from the West Bengal University of Technology, India, in 2009, and the M.S. and Ph.D. degrees in computer engineering from the University of Central Florida, Orlando, in 2011 and 2015, respectively. He is currently a Post-Doctoral Research Fellow with the CReWMAN Laboratory, Missouri University of Science and Technology. His research interests include anomaly detection, trust, reputation, and information assurance in dynamic spectrum access networks and cyber physical systems. He was a recipient of the IEEE PIMRC Best Paper Award. He also serves as a TPC Member of IEEE ICNC, IFIP Wireless Days, ICIT and a Reviewer of several international conferences, peer-reviewed journals.

Saptarshi Debroy received the B.Tech. degree from the West Bengal University of Technology, India, in 2006, the M.Tech. degree from Jadavpur University, India, in 2008, and the Ph.D. degree in computer engineering from the University of Central Florida in 2014. He was a Post-Doctoral Fellow with the University of Missouri. He is an Assistant Professor with the Computer Science Department, Hunter College and the Graduate Center with the City University of New York. He has published over 30 conferences and journal papers. His current research interests include dynamic spectrum access, cloud security, Internet of things, and cyberinfrastructure. He was a recipient of the Best Paper Award at IEEE PIMRC 2011. He serves as a TPC Member at conferences such as, IEEE ICC, IEEE ANTS, IEEE INFOCOM BigSecurity, ICDCN, and ICIT. He also serves as the Publicity Co-Chair in conferences, such as ACM MoViD and ICDCN.

Mainak Chatterjee received the B.Sc. (Hons.) degree in physics from the University of Calcutta, the M.E. degree in electrical communication engineering from the Indian Institute of Science, Bengaluru, and the Ph.D. degree from the Department of Computer Science and Engineering, University of Texas at Arlington. He is an Associate Professor with the Department of Computer Science, University of Central Florida, Orlando. His research interests include economic issues in wireless networks, applied game theory, cognitive radio networks, dynamic spectrum access, and mobile video delivery. He has published over 150 conferences and journal papers. He was a recipient of the Best Paper Awards at the IEEE Globecom 2008 and the IEEE PIMRC 2011, and the AFOSR Sponsored Young Investigator Program Award. He co-founded the ACM Workshop on Mobile Video. He serves on the Editorial Boards of Elsevier's *Computer Communications* and *Pervasive and Mobile Computing* journals. He has served as the TPC Co-Chair of several conferences. He also serves on the executive and technical program committees of several international conferences.