A Cooperative Spectrum Sensing Scheme for Cognitive Radio Ad Hoc Networks based on Gossip and Trust

Aida Vosoughi and Joseph R. Cavallaro
Department of Electrical and Computer Engineering
Rice University, Houston, Texas, USA

Alan Marshall
Department of Electrical Engineering and Electronics
University of Liverpool, Liverpool, UK

***Abstract*—In a cognitive radio ad hoc network, secondary users must cooperate in a decentralized way in order to determine the presence or absence of the primary user. In such a setting, malicious nodes deteriorate the cooperative spectrum sensing performance by reporting incorrect sensing information to the other nodes. We classify distributed cooperative spectrum sensing in cognitive radio ad hoc networks into two categories: consensus-based and non-consensus-based. We investigate and compare the sensitivity of these categories to spectrum sensing data falsification attacks and analyze the benefit of trust management in enhancing the performance of these methods. To this end, we introduce a novel trust-aware gossip-based scheme for distributed sensing. Our simulation results show that the proposed scheme significantly improves the cooperative sensing performance in the presence of malicious nodes in the network.**

I. Introduction

Dynamic spectrum access techniques enable the use of licensed frequency bands on an opportunistic non-interference basis where the secondary users (SU) must avoid interfering with the primary user (PU) of the channel [1]. Therefore, in cognitive radio networks, SU's must sense the channel periodically to accurately determine whether the PU is using the channel or not. The goal is to minimize detections errors (i.e. misdetections and false alarms).Cooperative spectrum sensing has been shown to be effective in relaxing the sensitivity requirements on individual secondary users and improving the overall sensing performance [1], [2], [3]. However, selfish or malicious nodes may exploit the cooperation by sharing incorrect sensing data. This type of mis-behavior is known as Spectrum Sensing Data Falsification (SSDF) attack [4] and has been studied in the literature [5], [6], [7], [8], [9]. Nevertheless, the majority of the existing literature is focused on resisting malicious behavior in *centralized* spectrum sensing where a base-station (fusion center) makes the final sensing decision for the SU network (such as in IEEE 802.22, wireless regional area networks working in TV white-spaces [10].) However, centralized cooperation has known shortcomings: The fusion center becomes a single point of failure and a communication bottleneck. Therefore such schemes are not scalable and they suffer from degraded reliability [11], [12].

In this paper, we focus on distributed (decentralized) cooperative spectrum sensing (DCSS) for Cognitive Radio Ad hoc Network (CRAHN) and make the following contributions: We classify existing DCSS schemes for CRAHN into consensus-based and non-consensus-based. We analyze and compare the sensitivity of these two categories to SSDF attacks and explore the impact of trust management in both types. We propose a novel trust-aware gossip-based DCSS that uses push-sum protocol and significantly improves the PU detection performance in the presence of SSDF.

II. Classification of Existing DCSS Schemes

We categorize DCSS schemes into consensus-based and non-consensus-based. In the latter, SU's have no intention to reach a consensus in the network. For example in a local-fusion scheme proposed by Chen et al. [5], each SU makes its final decision solely based on a single observation of its one-hop neighbors' reports. The underlying strategy in this scheme is similar to the centralized cooperative sensing (e.g. [13]) with the difference that in the distributed version each SU becomes its own fusion center. A known mitigation technique against SSDF attacks in this category is that each node assigns history-based trust scores to its neighbors and it weights their reports according to the scores [14]. Chen et al. [5] propose a weighted sequential probability ratio test with reputation scores. A diffusion strategy for DCSS is introduced by Yu et al. [12] where an iterative adapt-then-combine algorithm is used to estimate PU signal power. This scheme can be highly vulnerable to SSDF since if an SU pairs up with a malicious node, it will be affected greatly by the falsified measurement. Nevertheless, SSDF in diffusion schemes has not been studied yet and no resiliency solution against these attacks exists. The effect of selfish nodes and collusions on diffusion algorithms have been investigated in [15] and [16].

In consensus-based schemes, SU's try to reach a consensus among the whole network. For example, this can be done by distributed computing of the aggregate average of nodes' decisions (or measurements) in the network. Two main subcategories of consensus-based algorithms are gossip-based [17] and Laplacian-based algorithms [18], [19]. Ahmed et al. [3] propose a DCSS approach that uses Flajolet-Martin aggregation [20] and a gossip protocol for reaching consensus about PU presence. However, the authors do not consider the potential malicious behavior and therefore their proposed scheme is vulnerable to SSDF attacks. Li et al. [9] propose a Laplacian-based consensus algorithm for DCSS and [21] proposes a simple mitigation against SSDF that includes detecting outliers.

To the best of our knowledge, no existing work compares the performance of the two above-mentioned categories under SSDF attacks. In addition, while malicious behavior and trust in local-fusion schemes have been the focus of several studies [5], [13], [7], [6], SSDF attacks and trust management have not been studied sufficiently in consensus-based schemes. Therefore, it is important to investigate the resilience of these schemes against SSDF attacks with and without trust management. In fact, consensus-based schemes are by nature more sensitive to malicious behavior. Our simulation results in Section VI confirm this statement. This is because in a consensus-based approach without trust management, malicious data can easily propagate through the network, while in a local-fusion scheme there is a better chance that the malicious data is confined locally. On the other hand, with trust management, consensus-based approaches are expected to perform better than local-fusion because of global cooperation.

In Section V we propose a new consensus-based DCSS scheme based on gossip protocol [17] that integrates trust management and is resilient to SSDF. We compare our proposed scheme against a local-fusion approach which is inspired by [13] and [5] and is described in Section IV.

III. System Model

We consider a location area within the range of a single PU transmitting signal with power P_t. A set of M SU nodes are uniformly spread in the location area and move according to a random way point model [22]. The SU's have power detectors and detection is modeled as a binary hypothesis testing problem: H_0 if PU is absent and H_1 if PU is present [1]. The spectrum sensing performance of an SU is characterized by the probability of false alarm (P_{FA}) and misdetection (P_{MD}):

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

where Γ and γ denote the received power and the detection threshold, respectively. Generally, the goal is to minimize P_{MD} while maintaining an upper-bounded P_{FA} [23], [24]. For example IEEE 802.22 standard recommends fixing P_{FA} to 1% or 10% and derive the power detection threshold to evaluate P_{MD}. We follow the same framework in this paper. The received signal by an SU can be modeled as follows:

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

where $s(n)$ and $w(n)$ are the signal component with power P_s and the zero-mean additive white Gaussian noise with noise power P_n, respectively. If the power detector takes N samples, the test statistic is given by: $\Gamma = \frac{1}{N} \sum_{n=1}^{N} y(n) y(n)^*$. Using the central limit theorem, it can be shown that for large enough N, the test statistic follows a normal distribution [23]:

$$\Gamma \sim \mathcal{N}(P_s + P_n, \frac{2(P_s + P_n)^2}{N}) \tag{1}$$

False alarms occur when H_0 is true; therefore, in (1) $P_s = 0$. Thus, the power detection threshold (γ) can be derived from P_{FA} as follows ($Q(.)$ is the inverse Gaussian Q-function):

$$\gamma = P_n(1 + \sqrt{\frac{2}{N}} Q^{-1}(P_{FA})) \tag{2}$$

Hata (suburban areas variant) is used as the path loss model [25], [26]. The Hata model has been recommended by IEEE 802.22 working group for spectrum sensing modeling [5]. For small scale path loss variability, log-normal shadowing model is applied. The overall path loss (in dB) is given by $PL(d) = \overline{PL}(d) + X_\sigma \quad [dB]$, [27] where $\overline{PL}(d)$ is the average path loss as a function of transmitter-receiver distance d based on the Hata model, and X_σ is a zero-mean Gaussian random variable in dB with standard deviation σ in dB. In this model, each SU performs the detection by comparing its received signal power of $P_t - PL(d) + P_n$ with the threshold (γ).

IV. Non-Consensus-Based DCSS

This section describes a trust-aware non-consensus-based DCSS scheme inspired by the proposed local-fusion approaches in [5] and [13]. Algorithm 1 lists the steps each SU node i executes at time step t. Each SU i can communicate with its one-hop neighbors set denoted by N_i via a control channel. The final decision of a node is essentially driven by a weighted average of received local sensing reports from neighbors. Once a node makes its decision based on its own measurement, it performs a single local fusion of its neighbors' reports to derive the final decision. The local and final sensing decisions of node i at time t are denoted by $l_i(t), f_i(t) \in \{+1, 0, -1\}$ respectively, where $+1$ means the PU is present, -1 means the PU is absent, and 0 means uncertain. Note that this scheme does not follow a diffusion strategy [12]. A diffusion algorithm requires nodes to sense and update their measurements iteratively in each time step. In Algorithm 1, θ_{ij} denotes the trustworthiness score that node i assigns to its neighbor j based on its observations from j in their previous interactions (θ_{ii} is node i's self assessment). Note that trust management can be disabled by replacing all of θ parameters by 1. An SU calculates the trust score by keeping track of the number of positive and negative observations from a neighbor. The details of trust score calculation are given in Section V-A. In Algorithm 1, $sgn(.)$ is the sign function.

V. Proposed Trust-Aware Gossip-Based DCSS

We introduce a novel consensus-based DCSS scheme which is resilient to SSDF attacks. Our proposed scheme is based on push-sum protocol introduced in [17]. In push-sum, each node in the network holds a value and the goal is to calculate the average of these values using gossip. Gossip protocols are a class of distributed algorithms where at each time step each node transmits its value to a random target node. They are known to converge exponentially fast, on the order of $log(M)$ where M is the number of nodes in the network [17].

Each sensing time step (t), includes A aggregation rounds. In all rounds a, each node i maintains a sum $s_{a,i}$ and a weight $w_{a,i}$. These parameters are initially set as $s_{1,i} = l_i(t)$ and $w_{1,i} = 1$. At $a = 1$, node i sends the pair $(s_{1,i}, w_{1,i})$ to itself and at each consequent aggregation round $1 < a \leq A$ each

Algorithm 1: Non-consensus-based DCSS with trust management. Each node i executes this algorithm at time t.

Node i makes local sensing decision: $l_i(t)$;
i sends $l_i(t)$ to all of its neighbors and receives $l_j(t)$ from all neighbors $j \in N_i$;
i makes final decision: $f_i(t) = sgn(\frac{\theta_{ii}(t)\times l_i(t)+\sum_{j\in N_i}(\theta_{ij}(t)\times l_j(t))}{\theta_{ii}(t)+\sum_{j\in N_i}\theta_{ij}(t)})$;
for *each of i's neighbors* $j \in N_i$ **do**
    θ_{ij} = updated score considering new observation;
end

Algorithm 2: Trust-aware gossip-based DCSS. Each node i executes this algorithm at time t.

for *All aggregation rounds* $a = 2$ *to* A **do**
    Let (s_j, w_j) be the set of pairs that node i received in aggregation round $a-1$;
    i computes: $s_{a,i} = \sum_j \theta_{ij}(t)s_j, \quad w_{a,i} = \sum_j \theta_{ij}(t)w_j$;
    i chooses a neighbor k from the set N_i at random and sends the pair $(\frac{1}{2}s_{a,i}, \frac{1}{2}w_{a,i})$ to k and to itself;
end
Node i makes final decision: $f_i(t) = sgn(\frac{s_{A,i}}{w_{A,i}})$;
for *each of i's neighbors* $j \in N_i$ **do**
    θ_{ij} = updated score considering new observation;
end

node i performs the steps listed in Algorithm 2. At each round, node i's estimate of the average is $\frac{s_{a,i}}{w_{a,i}}$ and as the aggregation progresses, the estimate becomes more accurate and closer to the global average (consensus).

In order to make push-sum resilient to SSDF attacks for DCSS, we incorporate trust scores into the original algorithm. At each time step every node updates the history-based trust scores to its neighbors (the details of calculating the trust scores are discussed in the next subsection). At each aggregation round, node i multiplies each received pair (s_j, w_j) from neighbor j by node j's score (θ_{ij}). Therefore, if i believes that node j is trustworthy, then the report from j will have a high weight and otherwise it will have a low impact on the aggregation. Ideally every node's goal is to detect the untrustworthy neighbors and ignore the reports from them in their aggregate average by assigning them a zero trust score. Following this strategy, the more trusted nodes get a higher impact on the aggregate average of the network and the malicious nodes get lower weight or zero weight.

A. Trust Score Calculation and Update

The trust score calculation method that we use for both consensus-based and non-consensus-based schemes is inspired by [5] and [13]. At each time step t (when a sensing round is complete), node i may make a new binary observation from node j denoted by $o_{ij}(t)$:

$$o_{ij}(t) = \begin{cases} 1 & \text{if } g_{ij}(t) = f_i(t) \\ 0 & \text{if } g_{ij}(t) \neq f_i(t) \end{cases}$$

where $g_{ij}(t)$ denotes the last sensing report that node i received from node j and $f_i(t)$ is node i's final decision in this time step. Therefore, if j's report is in agreement with i's final decision, the observation is 1 and otherwise it is 0.

We propose that each node i maintains a binary observation vector per neighbor j denoted by O_{ij}. When node i makes a new observation from node j, it records it in the vector O_{ij}. The observation vector grows until it hits a predefined maximum size. From this point forward, the vector follows a FIFO structure and each time a new observation is made, the oldest observation is discarded and the new observation is pushed into the end of the queue. The trust score can be calculated as follows:

$$\theta_{ij}(t) = \frac{H(O_{ij})}{|O_{ij}|}$$

where $H(.)$ denotes the Hamming weight of the binary vector O_{ij} and $|O_{ij}|$ is the current size of the vector. In addition, in order to assign a trust score, node i must make enough number of observations from node j, that is if $|O_{ij}| < O_{min}$ then $\theta_{ij}(t) = 0$. Also if $\frac{H(O_{ij})}{|O_{ij}|} < 0.5$ we assign $\theta_{ij}(t) = 0$.

It is important to note that since there is no feedback from PU or any other authority about the presence or absence of PU in a sensing round, there is essentially no "ground truth" to rely on. Therefore, the best strategy for each honest SU is to rely on its own final decisions because the SU is at least certain about its own honesty, even though it does not know how reliable it is. Each node gradually build trust in its neighbors based on the degree of agreement between them. Nodes completely distrust new neighbors (if $|O_{ij}| < O_{min}$) and neighbors with whom the agreement is below 50%. Obviously, if an honest node is unreliable and constantly makes wrong decisions, it will always assign higher trust scores to unreliable or malicious nodes who agree with it and lower scores to reliable honest nodes. Therefore, in an all-honest but unreliable SU network, we anticipate that the trust integration degrades the DCSS performance. However, we expect these inevitable trust management errors to be small in contrast with the performance improvement of trust management driven by enforcing malicious nodes' low scores. Simulation results in Section VI confirm this analysis.

VI. Simulation Results and Analysis

As discussed in Section III, for evaluation purpose we fix a maximum $P_{FA} = 0.1$ for individual SU's and derive the required detection threshold from (2). Through Monte Carlo simulations, we derive P_{MD} as the average portion of the honest nodes in the network that misdetect in a sensing round. We measure P_{MD} in different DCSS schemes and in the presence of a variable number of malicious always-no nodes (nodes that constantly report PU absence to cause misdetection). The simulation parameters and corresponding values are listed in Table I.

Figure 1 compares P_{MD} of non-consensus-based and the proposed consensus-based schemes for a variable fraction of always-no nodes in the network (#Always-No/M). As a reference, we also show the average P_{MD} for single independent nodes in a non-cooperative sensing scenario (about 0.15 from Monte Carlo simulations). Obviously, the non-cooperative performance is independent of the number of malicious nodes.

TABLE I
SIMULATION PARAMETERS

Path Loss Model		Way Point Mobility Model		Noise Power		Others	
PU Distance from CRAHN	20Km	Min Velocity	1m/s	Noise Figure	11dB	# SU Nodes	50
PU Antenna Height	20m	Max Velocity	3m/s	Channel Bandwidth	6MHz	SU Network Area	250m×250m
SU Antenna Height	1m	Min Pause	120s	Ultimate Noise Floor	-174dBm	SU Node Range	60m
Center Frequency	615MHz	Max Pause	360s	P_{FA}	0.1	Sense Interval	2s
Log-normal Shadowing Standard Deviation	11.8dB	Prob(Static)	0.3	Noise Power	-95.22dBm	Min Observ. Window	2
Transmit Power	44 to 56 dBm			Detection Threshold	-94.88dBm	Max Observ. Window	10

Note that in our setup, on average each node has 8 neighbors, therefore the consensus-based scheme with 8 aggregation rounds has about the same communication overhead as the non-consensus-based. A higher number of aggregation rounds imposes higher communication requirements for the cooperative sensing. Figure 1 shows that our proposed trust-aware gossip-based DCSS outperforms the non-consensus scheme and increasing the number of aggregation rounds boosts the performance. The reason is that in a consensus-approaching scheme each SU node is virtually communicating globally with the whole network and not its local neighbors only. In addition, the results confirm that, as anticipated, the consensus-based scheme is more sensitive to SSDF attacks but our proposed trust integration significantly improves the performance and resilience to malicious nodes. For example, when 30% of the nodes are malicious, the proposed trust management decreases P_{MD} from 23% (in consensus-based, without trust, 8 aggregation rounds) to 5%.

In both consensus-based and non-consensus-based, the incorporation of trust makes the sensing strategy much more resilient to incorrect reporting from malicious nodes. Moreover, as the percentage of malicious nodes increases, the performance gap between "with trust" (solid curves) and "without trust" (dashed curves) becomes more significant. The curves corresponding to "without trust" schemes show that if more than 30% of the nodes are malicious, the cooperation becomes completely useless as the performance is worse than non-cooperating scenario. In contrast, trust management adds a significant extra tolerance to both consensus-based and non-consensus-based sensing schemes.

Figure 2 compares the performance of the sensing schemes in an all-honest but unreliable network, where there is no malicious node, but nodes can still report incorrectly due to, for example, poor reception. In order to evaluate for a variable degree of node unreliability, we run the simulations for variable transmit power. As discussed in Section V-A, for an all-honest network, there is a performance degradation due to trusting errors, however, the degradation is not significant and in the worst case (for high unreliability, on the right end of the curves) remains below 3% in all consensus-based variants. The performance degradation is greater in non-consensus scheme. As seen from Figure 2, the proposed consensus-based scheme outperforms the non-consensus method in an all-honest network as well.

VII. CONCLUSIONS

We explored two categories of decentralized cooperative spectrum sensing schemes for cognitive radio ad hoc networks:

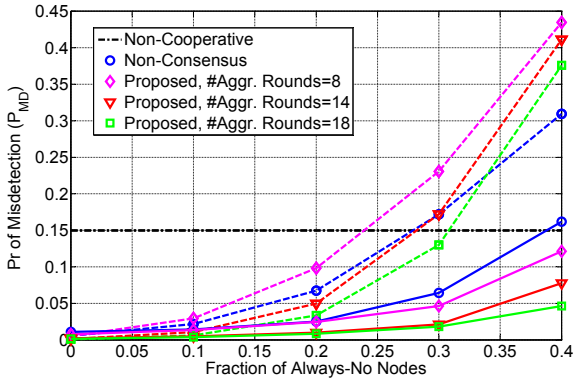

Fig. 1. Comparing probability of misdetection in the proposed consensus-based vs. non-consensus-based scheme in the presence of always-no nodes, $P_t = 52$dBm, solid (dashed) lines: with (without) trust management.

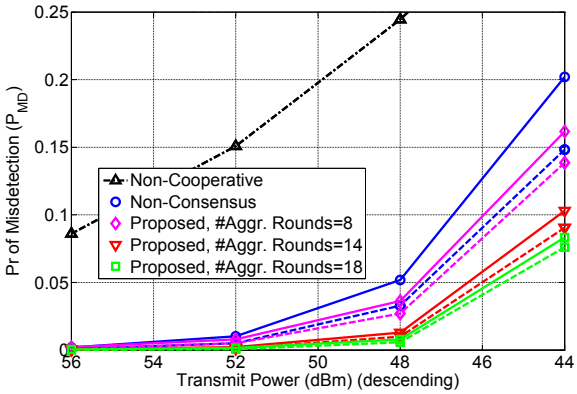

Fig. 2. Comparing probability of misdetection in the proposed consensus-based vs. non-consensus-based scheme in an honest network, solid (dashed) lines: with (without) trust management.

consensus-based and non-consensus-based. The consensus-based scheme is shown to outperform the non-consensus-based method. However, the results show that consensus-based schemes are more sensitive to SSDF attacks and therefore it is crucial to equip them with appropriate mitigation techniques. Our proposed trust management for gossip-based spectrum sensing significantly improves the detection performance in the presence of malicious nodes. As a future work, we plan to analyze the convergence speed of our proposed consensus-based sensing scheme in different scenarios and study the impact of observation vector size in trust management for cooperative sensing.

ACKNOWLEDGMENT

This work was supported in part by the US National Science Foundation under grants ECCS-1408370, CNS-1265332, and ECCS-1232274.

REFERENCES

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

[2] S. M. Mishra, A. Sahai, and R. W. Brodersen, "Cooperative sensing among cognitive radios," in *IEEE International Conference on Communications (ICC)*, vol. 4, 2006, pp. 1658–1663.

[3] N. Ahmed, D. Hadaller, and S. Keshav, "Guess: Gossiping updates for efficient spectrum sensing," in *Proceedings of the 1st International Workshop on Decentralized Resource Sharing in Mobile Computing and Networking*, ser. MobiShare '06. New York, NY, USA: ACM, 2006, pp. 12–17.

[4] R. Chen, J.-M. Park, Y. Hou, and J. Reed, "Toward secure distributed spectrum sensing in cognitive radio networks," *IEEE Communications Magazine*, vol. 46, no. 4, pp. 50–55, April 2008.

[5] R. Chen, J.-M. Park, and K. Bian, "Robust distributed spectrum sensing in cognitive radio networks," in *INFOCOM. The 27th Conference on Computer Communications*, April 2008, pp. 31–35.

[6] P. Kaligineedi, M. Khabbazian, and V. Bhargava, "Secure cooperative sensing techniques for cognitive radio systems," in *IEEE International Conference on Communications (ICC)*, May 2008, pp. 3406–3410.

[7] W. Wang, H. Li, Y. Sun, and Z. Han, "Catchit: Detect malicious nodes in collaborative spectrum sensing," in *Proceedings of the 28th IEEE Conference on Global Telecommunications*, ser. GLOBECOM'09, 2009, pp. 5071–5076.

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

[9] Z. Li, F. R. Yu, and M. Huang, "A distributed consensus-based cooperative spectrum-sensing scheme in cognitive radios," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 1, pp. 383–393, 2010.

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

[11] C. Lopes and A. Sayed, "Diffusion least-mean squares over adaptive networks: Formulation and performance analysis," *Signal Processing, IEEE Transactions on*, vol. 56, no. 7, pp. 3122–3136, July 2008.

[12] C.-K. Yu, M. van der Schaar, and A. Sayed, "Distributed spectrum sensing in the presence of selfish users," in *Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP), 2013 IEEE 5th International Workshop on*, Dec 2013, pp. 392–395.

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

[14] G. Baldini, T. Sturman, A. Biswas, R. Leschhorn, G. Godor, and M. Street, "Security aspects in software defined radio and cognitive radio networks: A survey and a way ahead," *IEEE Communications Surveys Tutorials*, vol. 14, no. 2, pp. 355–379, 2012.

[15] C.-K. Yu, M. van der Schaar, and A. Sayed, "Reputation design for adaptive networks with selfish agents," in *Signal Processing Advances in Wireless Communications (SPAWC), 2013 IEEE 14th Workshop on*, June 2013, pp. 160–164.

[16] C.-K. Yu, M. Laghate, A. Sayed, and D. Cabric, "On the effects of colluded statistical attacks in cooperative spectrum sensing," in *Signal Processing Advances in Wireless Communications (SPAWC), 2013 IEEE 14th Workshop on*, June 2013, pp. 275–279.

[17] D. Kempe, A. Dobra, and J. Gehrke, "Gossip-based computation of aggregate information," in *Foundations of Computer Science, 2003. Proceedings. 44th Annual IEEE Symposium on*, Oct 2003, pp. 482–491.

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

[19] R. Zhou and K. Hwang, "Gossip-based reputation aggregation for unstructured peer-to-peer networks," in *IEEE International Parallel and Distributed Processing Symposium*, 2007, pp. 1–10.

[20] P. Flajolet, G. N. Martin, and G. N. Martin, "Probabilistic counting algorithms for data base applications," 1985.

[21] H. Tang, F. R. Yu, M. Huang, and Z. Li, "Distributed consensus-based security mechanisms in cognitive radio mobile ad hoc networks," *IET communications*, vol. 6, no. 8, pp. 974–983, 2012.

[22] C. Bettstetter, G. Resta, and P. Santi, "The node distribution of the random waypoint mobility model for wireless ad hoc networks," *Mobile Computing, IEEE Transactions on*, vol. 2, no. 3, pp. 257–269, July 2003.

[23] S. J. Shellhammer, "Spectrum sensing in IEEE 802.22," *IAPR Wksp. Cognitive Info. Processing*, pp. 9–10, 2008.

[24] G. C. M. M. Steve Shellhammer, Victor Tawil and M. Ghosh, "Spectrum sensing simulation model, ieee 802.22-06/0028r10."

[25] M. Hatay, "Empirical formula for propagation loss in land mobile radio services," *Vehicular Technology, IEEE Transactions on*, vol. 29, no. 3, pp. 317–325, Aug 1980.

[26] A. F. Molisch, *Wireless communications*. John Wiley & Sons, 2010, vol. 15.

[27] T. S. Rappaport, *Wireless communications: principles and practice*. Prentice Hall PTR New Jersey, 1996, vol. 2.