Enhanced Robust Cooperative Spectrum Sensing in Cognitive Radio

Feng Zhu and Seung-Woo Seo

Abstract: **As wireless spectrum resources become more scarce while some portions of frequency bands suffer from low utilization, the design of cognitive radio (CR) has recently been urged, which allows opportunistic usage of licensed bands for secondary users without interference with primary users. Spectrum sensing is fundamental for a secondary user to find a specific available spectrum hole. Cooperative spectrum sensing is more accurate and more widely used since it obtains helpful reports from nodes in different locations. However, if some nodes are compromised and report false sensing data to the fusion center on purpose, the accuracy of decisions made by the fusion center can be heavily impaired. Weighted sequential probability ratio test (WSPRT), based on a credit evaluation system to restrict damage caused by malicious nodes, was proposed to address such a spectrum sensing data falsification (SSDF) attack at the price of introducing four times more sampling numbers. In this paper, we propose two new schemes, named enhanced weighted sequential probability ratio test (EWSPRT) and enhanced weighted sequential zero/one test (EWSZOT), which are robust against SSDF attack. By incorporating a new weight module and a new test module, both schemes have much less sampling numbers than WSPRT. Simulation results show that when holding comparable error rates, the numbers of EWSPRT and EWSZOT are 40% and 75% lower than WSPRT, respectively. We also provide theoretical analysis models to support the performance improvement estimates of the new schemes.**

Index Terms: **Cognitive radio (CR), cooperative spectrum sensing, enhanced weighted sequential probability ratio test (EWSPRT), enhanced weighted sequential zero/one test (EWSZOT), spectrum sensing data falsification (SSDF) attack, weighted sequential probability ratio test (WSPRT)**

I. INTRODUCTION

Wireless frequencies are scarce resources and their usage is strongly regulated by government agencies. Most of the bands are allocated to licensed users for various purposes, and are restricted from unlicensed users. However, a large portion of the assigned spectrum is used very inefficiently because usage is concentrated in certain portions of the spectrum. Data from the Federal Communications Commission (FCC) say that the spectrum utilization rate can vary from 15% to 85% [1]. With the dramatic increase in demand for spectrum resources from various newly-emerged wireless networks in recent years, the fixed spectrum assignment policy is creating more problems, and lack of spectrum will restrict the development of future wireless systems.

The limited available spectrum and the inefficiency in spectrum usage motivate the idea that unlicensed users could utilize the existing wireless spectrum opportunistically in little-used or idle bands. Cognitive radio (CR) is a new communication system adopting this idea to ease current spectrum inefficiency problems. It provides unlicensed users with the capability to use or share the licensed band in an opportunistic manner, as long as they do not bring serious interference to licensed users. Specifically, CR technology will enable unlicensed users to determine which portions of the spectrum are available by detecting the presence of licensed users, to select the best available channel, to coordinate access to the channel with other unlicensed users, and to quit the channel when a licensed user is newly detected.

A CR network is composed of primary users and secondary users. The primary user defines the user having exclusive right to a certain band and is only controlled by primary base station (BS). Primary users and primary BS constitute the primary network while secondary users and secondary BS constitute the secondary network. A secondary user is a user trying to have opportunistic usage of the licensed bands. In the CR system, secondary users are responsible for coexistence while there should be no modification of hardware and software required by primary users.

Security is one of the crucial research topics in CR because of the importance of CR network reliability. In the security field, an attack on CR can be defined as any activity that results in unacceptable interference to primary users or missed opportunities for secondary users. An attack is considered strong if it involves a small number of adversaries performing few operations while causing extensive damage to the network. Attacks on CR can be classified by the various layers from which they are launched. For example, in the PHY layer, an adversary can launch the jamming attack [2] by continuous transmission. An Overlapping attack [2] will harm another CR network if two CR networks overlap. In the MAC layer, the asynchronous sensing attack [3] can disturb normal sensing operation by purposefully transmitting in a sensing period. In cooperative spectrum sensing, to examine the existence of the primary user, a node takes its neighbors' sensing results as a reference in addition to its own sensing. If the compromised node reports false sensing data to its neighbors on purpose, it causes a spectrum sensing data falsification (SSDF) attack. The SSDF attack is a serious attack in CR which can disturb normal cooperative spectrum sensing and cause dysfunction in the network. Weighted sequential probability ratio test (WSPRT) is the only scheme robust against this attack; however, it requires high sampling numbers.

In this paper, we propose two new schemes to solve the sampling overhead problem in previous schemes. The main contribution of our work can be summarized as follows:

Manuscript received October 5, 2008.

The authors are with the Seoul National University, Korea, email: {zhufeng, sseo}@snu.ac.kr.

1229-2370/09/$10.00 © 2009 KICS

1. We propose two new schemes which are strong against SSDF and have much less sampling numbers than WSPRT. The first scheme, named enhanced weighted sequential probability ratio test (EWSPRT), uses a new weight module and the same test module as WSPRT, yet outperforms WSPRT by 40%. The second scheme, named enhanced weighted sequential zero/one test (EWSZOT), uses a new weight module and a new test module, and can outperform EWSZOT by 75%.
2. None of the previous works can provide a model of the sampling number of their scheme. Here, we propose a mathematical model for these schemes and the analytical results closely match the simulation results.

The rest of the paper is organized as follows. Section II provides background on cooperative spectrum sensing, SSDF attacks and the WSPRT scheme. Section III describes two new schemes, EWSPRT and EWSZOT, in detail. A mathematical model for sampling overhead is introduced in Section IV. Performance analysis results are given in Section V. Finally, our conclusions are drawn in Section VI.

II. Background: Cooperative Spectrum Sensing, SSDF Attack, and WSPRT

Spectrum sensing is the fundamental function of cognitive radio because only correct sensing provides the chance of potential usage without interference. Cooperative spectrum sensing has better sensing accuracy than non-cooperative sensing and is more widely used. However, a SSDF attack will dramatically impair normal sensing and cause failure of the network. Until now, WSPRT is the first and only scheme robust against SSDF attack in spite of very high sampling nunmbers.

A. *Cooperative Spectrum Sensing*

A CR system should be designed to be aware of, and sensitive to, the changes in its surroundings. Spectrum sensing enables the CR system to adapt to its environment by detecting and utilizing spectrum holes.

Although the most efficient way to detect spectrum holes is to detect the primary users receiving data within the communication range of a secondary user, the difficulty in locating a primary receiver focuses most research work on how a secondary user can detect the signal from a primary transmitter instead of a primary receiver.

In general, spectrum sensing techniques can be classified as non-cooperative sensing and cooperative sensing.

In noncooperative sensing, each node senses and judges seperately. Noncooperative sensing can be divided into three subcategories: Matched filter detection, energy detection, and cyclostationary feature detection. Matched filter detection [4] is the optimal choice in a stationary Gaussian noise environment if the information of a primary user is known. It requires less sensing time, although it demands a priori knowledge such as modulation type, packet format, etc. Energy detection [4] is useful if sufficient information cannot be gathered. The signal energy level is compared with a threshold for judging the existence of a primary user. Although simple since no priori knowledge is required, the performance is suspect and it is hard to differentiate unknown signal types. Cyclostationary feature detection [5] uses the cyclostationarity feature; i.e., the periodicity of internal mean and autocorrelation of periodic signals, to differentiate noise energy from signal energy, at a cost of long time and computational complexity.

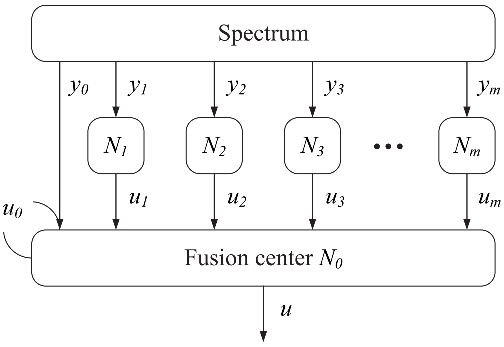

Fig. 1. Data fusion model.

On the other hand, cooperative spectrum sensing involves multiple nodes when deciding. Locally, each node utilizes the previously described method like matched filter detection. However, the node of interest refers to neighbors' results as well to make the final judgement. Noncooperative spectrum sensing is regarded as less accurate than cooperative spectrum sensing due to its failure to deal well with weak and unstable signals and its possible failure to solve the hidden terminal problem. Through obtaining helpful reports from neighbors, cooperative sensing mitigates the uncertainty in single detection as well as the multipath fading and shadowing effects. Cooperative sensing can be implemented either in a centralized or in a distributed manner. In the centralized method, the secondary BS plays the role of gathering all sensed information from the secondary users and making the decision. In the distributed mode, each node plays the roles of both a decision maker and a reporter.

B. *Data Fusion Model in Cooperative Spectrum Sensing*

When cooperative spectrum sensing is used, it is necessary to discuss how the collected sensing results from the node of interest, and its neighbors are fused together and raise a final claim for the presence of the primary user. This can be modeled as a classic data fusion problem.

Fig. 1 [6] shows a classic data fusion model. N_0 is the fusion center, and N_1 to N_m are sensing terminals. y represents the channel status, and y_0 to y_m are the signals received by the terminals. u_1 to u_m are sensing results sampled by the fusion center. $u_i = 1$ stands for a primary user is detected and $u_i = 0$ stands for absence. u_i could be different from y_i due to many causes like channel noise or SSDF attack. The fusion center finally draws a global decision, u, based on sampled u_i's. $u = 1$ means the existence of a primary user and $u = 0$ means absence.

Depending on the different ways to deal with reported data of u_i's, several data fusion schemes have been proposed.

1. $0/1$ fusion [7] simply counts the number of ones reported from all the terminals. A threshold is pre-specified. If the sum of ones is no less than the threshold, it decides $u = 1$, otherwise $u = 0$. Some logical rules can also be realized by the careful design of the threshold. For example, the OR rule equals to a threshold value of one, the AND rule equals to a

value of $(m+1)$, and a value of $(m+1)/2$ is a majority rule.

2. The Bayesian test [8] requires a priori knowledge of the probabilities of u_i's under hypothesis zero and one, which can be denoted as $P(u_i|H_0)$ and $P(u_i|H_1)$. The knowledge of a priori probabilities of u is also required. There are four cases and each case is allocated a cost. For two correct sensing cases, the cost is small, while for the false negative and false positive cases, the cost is large. The overall cost is the weighted sum of all four cases and the test makes a final decision which minimizes the overall cost.
3. The sequential probability ratio test (SPRT) [9] requires $P(u_i|H_0)$ and $P(u_i|H_1)$. Two bounds, the lower bound λ_0 and the upper bound λ_1, are defined. Unlike the previous three paralleled tests where the sampling number is a certain definite value, SPRT executes the test sequentially and has a dynamic sampling number for each test. The samples are dealt with one-by-one and the test is terminated when the probability ratio meet either of two bounds. Compared to the other three schemes, SPRT takes the fewest samples due to its ability to jump out of the test after taking the minimum necessary samples.

C. SSDF Attack and WSPRT

One threat to cooperative sensing is malicious secondary users reporting false sensing data when sampled. An adversary may alter local spectrum sensing reports on purpose, misleading the fusion center to make the wrong decision. This type of attack is called SSDF. None of previously described fusion data techniques is robust against SSDF attack due to their common features of not tracing the reporting history from a specific terminal, which eases the SSDF attacker to constantly harm the network without penalty.

Until now, WSPRT [10] is the only scheme addressing SSDF attack. WSPRT can recognize the malicious node from its report history and only accepts samples from reliable sensing terminals. WSPRT is composed of two parts: A credit maintenance/weight allocation module, and a sequential hypothesis test module. In the credit/weight module, a terminal's credit is allocated based on the accuracy of its sensing. If its local sensing report is consistent with the global decision, its credit receives one point bonus, otherwise one point penalty. The weight is defined as the normalized credit, and is applied as the index of probability ratio in the test. The hypothesis test of WSPRT is the same as SPRT except for the implemented weight:

$$Y = \prod_{i=0}^{m} (\frac{P[u_i|H_1]}{P[u_i|H_0]})^{w_i}. \quad (1)$$

D. Drawbacks of WSPRT

Although WSPRT is secure against SSDF attack, it has several serious drawbacks.

First, WSPRT can only address an SSDF attack at the cost of much higher sampling numbers. Simulation in [10] shows that WSPRT sampling numbers reaches four to five times that of SPRT, which means the sensing time is also four to five times longer. Since spectrum sensing needs to be executed frequently and periodically in a real world system, this cost is non-trivial and could restrict it from wide use.

Second, the WSPRT algorithm does not carefully think through every possible case and only works in normal status. Under some unexpected status or extreme assumptions, it can easily become stuck, and can even deadlock. The instability of the system lowers the value of WSPRT.

Third, WSPRT treats the wireless environment quite simply and many parameters are fixed despite of dynamic environment, which allows the method to perform well only in a smooth and stable environment. This lack of flexibility prohibits the widespread usage of WSPRT under various circumstances.

Finally, without a mathematical model, simulation results are not enough to support the validity of WSPRT.

III. PROPOSED SCHEME: EWSPRT AND EWSZOT

In this section, we propose two new algorithms, both of which outperform WSPRT in sampling numbers. EWSPRT adopts a more efficient weight module while using the same test module as WSPRT. EWSZOT adopts the same new weight module as EWSPRT and uses a new sequential 0/1 test module instead of SPRT, which greatly simplifies the algorithm's complexity.

A. New Scheme 1: EWSPRT

With EWSPRT, we propose new features including aggressive weights, soft decision, the "best of rest" strategy and truncated test to provide a more efficient and stable performance. Periodic noise measurement is also used to address the dynamic wireless environment which WSPRT did not consider. These features enable EWSPRT to run a faster test than WSPRT, and are described as follows:

A.1 Aggressive Weight Allocation

The same as in WSPRT, after the initialization of the system, every node's credit is set to zero, and nodes can accumulate credits by correct reports. Whenever a node's report is consistent with the global decision, its credit is increased by one; otherwise decreased by one. Statistically, in the long run, a sensing terminal with more accurate reports will always have higher credit than a terminal with less accurate reports. If we denote the credit for node i by c_i, the credit system can be represented as:

$$c_i = \begin{cases} c_i + 1, & u_i = u, \\ c_i - 1, & u_i \neq u. \end{cases} \quad (2)$$

And we use w_i to denote weight of node i. As mentioned, weight is normalized credit and the one used in the test. In WSPRT, weight is normalized with the maximum credit. Here, we use a new weight allocation method wherein the weight is normalized with the average of credit:

$$w_i = f_1(c_i) = \begin{cases} 0, & c_i < -g, \\ \frac{c_i+g}{avg(c_i)+g}, & c_i > -g \end{cases} \quad (3)$$

where $avg(c_i)$ denote the average credit over all nodes. The same as in WSPRT [10], g is a small positive value, which allows good nodes to have a slightly negative credit value if it suffers consecutive reporting errors due to various reasons, though this possibility is very low. In [10], $g = 5$, but if g is an integer, for some nodes, the denominator $avg(c_i) + g$ could be zero. So here we set g as 5.51.

Thus the hypothesis test of EWSPRT can be expressed as:

$$W = \prod_{i=0}^{m} \left(\frac{P[u_i|H_1]}{P[u_i|H_0]} \right)^{f_1(c_i)}, \tag{4}$$

$$\begin{cases} W \geq \lambda_1 \Rightarrow \text{accept } H_1, \\ W \leq \lambda_0 \Rightarrow \text{accept } H_0, \\ \lambda_0 < W < \lambda_1 \Rightarrow \text{take another round} \end{cases}$$

where λ_1 is an upper bound and λ_0 is a lower bound.

Eqn. (3) works better than WSPRT because, as with the index of probability ratio, weight plays an important role in determining the speed of convergence. Normalization with maximum credit in WSPRT is too conservative, where only one node–the one having the maximum credit–reaches the weight of 1, while all other nodes can only have a weight less than 1. In fact, for SPRT which lacks any credit system, every node conceptually can be regarded to have a weight of 1. If n nodes are involved in cooperative sensing, the total sum of weights in SPRT and EWSPRT is n, whereas in WSPRT it is much less than n. Thus, if the same λ_0 and λ_1 are used in the test, EWSPRT and SPRT have the similar speed to convergence, while it would be much slower for WSPRT to meet either bound. The computation complexity of (4) is almost the same as WSPRT.

A.2 Soft Decision

A more aggressive weight could help increase the speed of the test, however it might introduce additional instability and increase error rate. The dynamic wireless environment creates problems of high uncertainty and vague detection. It could affect the accuracy more heavily if a node with high credit occasionally makes vague decisions.

Hard decision, which means the report always consists of one bit (1 or 0) as used in WSPRT, has the limitation of the existence of vague decisions. For example, assume that node A only has 55% certainty that 0 is detected, while node B has 90% certainty that 0 is detected. However, when reported to the fusion center, there is no difference between 0 with 55% certainty and 0 with 90% certainty, even though they should be treated differently to make more accurate decision.

In our EWSPRT scheme, soft decision is introduced to mitigate this problem. Soft decision, also called as multi-bit decision, uses extra bits to express more information. The fusion center, when aware of more information, can operate in a more intelligent and accurate manner.

In order to implement soft decision, the first step is to define the way to express side information. In EWSPRT, we utilize a simplest two-bit soft decision mechanism, where the extra bit is used to decribe the degree of confirmation on the report. Thus all reports are divided into four more-detailed categories, differentiating a strong confirmation of 0 and 1 with a weak confirmation of 0 and 1 separately:

$$u_i = \begin{cases} 1 & (strong) & x > \gamma_1, \\ 1 & (weak) & \gamma < x < \gamma_1, \\ 0 & (weak) & \gamma_0 < x < \gamma, \\ 0 & (strong) & x < \gamma_0. \end{cases} \tag{5}$$

Three signal thresholds γ_1, γ, and γ_0 ($\gamma_1 > \gamma > \gamma_0$) are used here: γ is used for differentiation of 0 and 1, γ_1 is used to differentiate the strong and weak confirmation of 1, and γ_0 is used to differentiate the strong and weak confirmation of 0.

The second step is to define how fusion center reacts to soft reports. In our EWSPRT, we want the full weight of the node to be applied if it presents a strong confirmed report. With a weak confirmed report, only half of the real weight is applied. By treating weight in a more conservative manner, EWSPRT can restrict the harm of vague reports. To do this, we define a new reaction function $f_2(u_i)$ as follows:

$$f_2(u_i) = \begin{cases} 1 & x > \gamma_1 & (strong\ 1), \\ 1/2 & \gamma < x < \gamma_1 & (weak\ 1), \\ 1/2 & \gamma_0 < x < \gamma & (weak\ 0), \\ 1 & x < \gamma_0 & (strong\ 0). \end{cases} \tag{6}$$

The fusion center will multiply $f_2(u_i)$ to the weight function $f_1(c_i)$ which is the index of probability ratio, and the new description of EWSPRT with soft decision is:

$$W = \prod_{i=0}^{m} \left(\frac{P[u_i|H_1]}{P[u_i|H_0]} \right)^{f_1(c_i)f_2(u_i)}, \tag{7}$$

$$\begin{cases} W \geq \lambda_1 \Rightarrow \text{accept } H_1, \\ W \leq \lambda_0 \Rightarrow \text{accept } H_0, \\ \lambda_0 < W < \lambda_1 \Rightarrow \text{take another round.} \end{cases}$$

Soft decision is much more helpful when energy detection is utilized in a system because in that case energy is the only factor in decision making. Thus, enabling the fusion center to be aware of the energy certainty could be quite crucial. On the other hand, if cyclostationary feature detection is used, as long as the signal strength is beyond the level of sensitivity, terminals have other information; i.e. cyclostationary features to differentiate primary signal from high power noise. In this case, side information provided by soft decision is less crucial.

Because the two-bit soft decision of EWSPRT only introduces extra overhead of one bit, it could be completely neglected in a modern wireless protocol.

A.3 Best of Rest Strategy

Smart arrangement of the polling order helps significantly in boosting performance. This intelligence is necessary since it is natural to give some nodes higher priorities. Nodes are distributed and keep moving in a wireless environment, and are greatly affected by a shadowing effect. A node in a good location could undergo more accurate measurement and hold better records while others might not. Among terminals of different brands and types, those with better sensitivity also hold better records. Generally, the credits reflect the capabilities of the nodes, and are consistent with their importance.

So, in EWSPRT, we demand that the fusion center hold a list of nodes in descending order and update it before every round of tests. In the upcoming polling, it should always request the node with the highest credit first, then turn to the second best node, the third best node etc.

WSPRT did not have any preference among nodes. The fusion center just randomly polls a node for its report. If the bound is not met it continues to randomly poll the next node. This increases the chance that nodes with lower credit, i.e., a small weight, could be polled earlier and slow the test significantly. Lack of an intelligent; i.e., only random polling, makes WSPRT less competitive than EWSPRT in reality.

Compared to WSPRT, EWSPRT additionally requires a list holding the order of node credit and a sorting algorithm. Considering the capabilities of modern devices, this load could be neglected.

A.4 Truncated Test

Truncated test is designed to enhance the stability as well as the efficiency of a system. Only in an ideal situation, W will always meet either bound and terminate the test after several rounds of polling. However, due to the fading effect and the coincident distance from a primary user to the terminals, it is common that the received signal is neither strong nor weak, leading to a probability ratio neither large enough nor small enough. In this case, even after many rounds, W may be still bouncing between two bounds and never jump out. The test may thus become very slow and may even become stuck in a deadlock.

For these reasons, we introduce a truncated test wherein a threshold $Round_{max}$ is pre-determined. After $Round_{max}$ reports have been taken, the test is forced to end.

After jumping out of the test, decision making is determined by policy. It coulde be configured to have a 50%-50% chance to decide 0 or 1. Or if a conservative policy defined, it always supposes the existence of a primary user when unsure. Considering the necessity of protecting the right of primary user in the cognitive radio, in our current design, conservative policy is used to minimize the interference to the primary user.

The combination of truncated test and best of rest policy have extra benefits since, after a reasonable number of rounds, the remaining nodes usually apply a quite small weight and play insignificant roles in the decision. Therefore not reflecting them in the decision could speed up the test with less harm to error rate performance.

A.5 Periodic Noise Measurement

The value of a priori probabilities in test can be decided empirically though it introduces great error. [10] mentioned a method to derive it from the node's received power. First, HATA model is adopted as the path loss in cognitive radio. With that, any node can derive its attenuated received power from BS. Here HATA model in rural area is used and specified as $\overline{P}(r) = P_t - 27.77 - 9.39\log f_c + 4.78(\log f_c)^2 + 3.82\log h_{te} + (1.1\log f_c - 0.7)h_{re} - (44.9 - 6.55\log h_{te})\log d$. Where P_r is the received power, transmit power $P_t = 85$ dbm, working frequency $f_c = 617$, height of transmitter $h_{te} = 200$ m, height of receiver $h_{re} = 1$ m, d is the distance between transmitter and receiver. Then, with the knowledge of received power and noise distribution, a priori probabilities can be deduced as $P_{11} = Q((\gamma - P_r)/\sigma_{fix})$, $P_{10} = Q((\gamma - n_{0fix})/\sigma_{fix})$, where P_{11} stands for the priori probability when local sensor reports 1 when the channel is 1, the definition of P_{10}, P_{01}, and P_{00} are similar. And channel noise is always assumed to be a fixed AGWN (n_0, σ), which is constant anytime and anywhere.

This method takes location and path loss effect into consideration, so it is much closer to the truth and more favorable than empirical values. However, if the noise feature in real world is far from its assumption, its error rates could also be very high. Unfortunely, in cognitive radio, this case might happen a lot because nodes keep moving into new places with totally new noise backgrounds. As time varies, other factors emerge to change the noise level, e.g. the appearance of an electronic device working in a near band. Thus, the probability ratios calculated under fixed noise power are very unreliable.

Therefore, in EWSPRT, we suggest periodic noise measurement to address the dynamic noise environment in cognitive radio. Terminals are required to measure the noise periodically. The measurement interval should be carefully selected such that the noise level is reflected properly and in a timely manner, without causing excessive system loads. The noise parameter newly detected at current time t, (n_{0_t}, σ_t) will be used for calculation in the next period. Thus, we have,

$$\begin{aligned} P_{11} &= Q(\tfrac{\gamma - P_r}{\sigma_t}), \quad P_{01} = 1 - P_{11}, \\ P_{10} &= Q(\tfrac{\gamma - n_{0_t}}{\sigma_t}), \quad P_{00} = 1 - P_{10}. \end{aligned} \tag{8}$$

Periodic noise measurement could brings some load to a system. However, if the terminal is not very busy, periodic measurement does not do much harm to other operations, and could be used to improve to report accuracy. Thus, there is a tradeoff between system resource and performance.

With all five features in the algorithm, we can describe EWSPRT as the following pseudo-code:

```
1: $\forall i, c_i = 0$. $\{t_i\} = 0$.
2: For each sensing round of Node $N_0$ {
3: $i = 0, W = 1$.
4: If $round > Round_{max}$, go to step 12. (Truncated test)
5: Sorting algorithm, $t_1 = j$ (Node of highest credit), $t_2 = k$
   ...
6: $N_{t_i}$ measures $(n_{0_t}, \sigma_t) \rightarrow P_{11}, P_{10}, P_{01}, P_{00}$. (Periodic
   noise measurement)
7: Get a report $u_{t_i}$ from $N_{t_i}$. (Best of rest)
8: $W = W \left(\frac{P[u_{t_i}|H_1]}{P[u_{t_i}|H_0]}\right)^{f_1(c_{t_i}) f_2(u_{t_i})}$ (Soft decision &
   aggressive weight)
9: If $\lambda_0 < W < \lambda_1$, $i = i + 1$. Go to step 4.
10: If $W > \lambda_1$, decide $u = 1$. Go to step 13.
11: If $W < \lambda_0$, decide $u = 0$. Go to step 13.
12: Decide $u = 1$ by conservative policy.
13: For each $N_i$, if $u_i = u$, $c_i = c_i + 1$; else $c_i = c_i - 1$
14: }
```

B. Proposed Scheme 2: EWSZOT

EWSZOT is also composed of a weight module and a test module. EWSZOT uses the same weight module as EWSPRT, whose detailed features have been described previously. The only difference is that since a sequential test is not implemented here, there is less need to utilize the soft decision function. So, in order to reduce complexity, it is excluded from the weight module features.

In a classic data fusion model, 0/1 fusion is only implemented in a parallel way, which is not efficient in sampling overhead. In addition, its rule (AND, OR, Majority) is too simple to meet the need for a flexible threshold. Hence, we propose the principle of our sequential 0/1 test (SZOT): A suitable threshold q is selected first, which defines an upper bound of q and a lower bound of $-q$. The samples are taken one-by-one to execute the sequential test, and the test is terminated when the difference between the number of reported 1 values and the number of reported 0 values meets the upper bound or the lower bound. SZOT can be represented as:

$$S = \sum_{i=0}^{m} (-1)^{u_i+1},$$
$$\begin{cases} S \geq q \Rightarrow \text{accept } H_1, \\ S \leq -q \Rightarrow \text{accept } H_0, \\ -q < S < q \Rightarrow \text{take another round.} \end{cases} \tag{9}$$

After combining the weight module and the test module, the EWSZOT test can be described as follows:

$$W = \sum_{i=0}^{m} (-1)^{u_i+1} w_i,$$
$$\begin{cases} W \geq q \Rightarrow \text{accept } H_1, \\ W \leq -q \Rightarrow \text{accept } H_0, \\ -q < W < q \Rightarrow \text{take another round.} \end{cases} \tag{10}$$

EWSZOT's merit over EWSPRT is its simplicity. Since SPRT is given up, there is no need to calculate the probability ratio based on complicated density distribution and HATA model everytime, which is the most time consuming part in the test. Hence, the complexity and thus the running speed of the test has been dramatically decreased. Considering that spectrum sensing is executed periodically and frequently in cognitive radio, this result is quite meaningful for practical usage.

IV. MATHEMATICAL MODEL

None of previous works was able to provide any mathematical model for the test. Currently only work on the modeling of the sampling number in SPRT can be found [12], which is too simple to describe these complicated tests. In this section, we significantly expand the work and develop a new model of the sampling number for tests with malicious nodes and with weights, which is suitable for WSPRT, EWSPRT, and EWSZOT.

A. *Existing Model for SPRT*

The classic sequential test textbook [12] provides a formula for the sampling number of SPRT. The total sampling number $E[Sam]$ is expected as:

$$E[Sam] = E[Sam|H_1]P(H_1) + E[Sam|H_0]P(H_0) \tag{11}$$

where $E[Sam|H_1]$ and $E[Sam|H_0]$ are sampling numbers under hypothesis 1 and hypothesis 0, $P(H_1)$ and $P(H_0)$ are the probability hypothesis 1 and hypothesis 0 happen separately. And

$$E[Sam|H_1] = \frac{E[L(W)|H_1]}{E[L(PR)|H_1]} = \frac{(1-b)\log\lambda_1 + a\log\lambda_0}{E[L(PR)|H_1]},$$
$$E[Sam|H_0] = \frac{E[L(W)|H_0]}{E[L(PR)|H_0]} = \frac{(1-a)\log\lambda_0 + b\log\lambda_1}{E[L(PR)|H_0]} \tag{12}$$

where a and b are defined as the false positive and false negative rate of the test, $\lambda_1$and λ_0 are two thresholds, $E[L(W)|H_1]$ and $E[L(W)|H_0]$ are the logarithmic expectation of W under H_1 and H_0. $E[L(PR)|H_1]$ and $E[L(PR)|H_0]$ are the logarithmic expectation of probability ratio under H_1 and H_0, respectively.

B. *Main process of modeling*

Our contribution is expanding the previous model to cases where m malicious nodes exist among n nodes, and where a weight module is used in test.

Three types of SSDF attacks are discussed in this paper: Always-false, always-busy, and always-free. The always-false attacker always reports the opposite value of its real spectrum sensing results. The always-busy attacker always claims the spectrum is busy while the always-free attacker always claims the spectrum is free.

We take always-free attack in EWSPRT as an example to describe the process of deriving in detail the average sampling number per node. The modeling for always-false attack, always-busy attack, EWSPRT, and EWSZOT are similar, as long as the related parts change correspondingly.

The model is composed of two parts. The first part models the weight module, and the second part models the test module and sampling number.

B.1 Modeling of Weight Module

First, we define $E(Dis)$, the average distance of a node from a BS when it is moving in an area. In the scope of this paper, a node has equal chance to appear at any location in the area, uniform distribution of its location is assumed, and there is $E(Dis) = \int_x \int_y \sqrt{(x-x_{BS})^2 + (y-y_{BS})^2} dxdy$. Similarly, according to the analysis of Section III.A.5, dynamic noise feature is also well distributed anywhere and anytime in the whole area, so here uniform distribution is also assumed to n_0 and σ. And we have the expectation of n_0 and σ, $E(n_0) = \int_x \int_y n_0(x,y)dxdy$ and $E(\sigma) = \int_x \int_y \sigma(x,y)dxdy$. According to (8), P_{11}, P_{01}, P_{10}, P_{00} are the functions of received signal strength. And according to HATA model, received signal strength is a fuction of $E(Dis)$. So we have

$$P_{10} = f[E(Dis), E(n_0), E(\sigma)], \quad P_{00} = 1 - P_{10}, \tag{13}$$
$$P_{11} = f[E(Dis), E(n_0), E(\sigma)], \quad P_{01} = 1 - P_{11}. \tag{14}$$

Here, for the convenience of analysis, we introduce the notion of function f, which is an abstract function and hides the detail of HATA model.

We define $P[c_g|H_1]$ as the probability a good node can get a credit increment per round when BS is busy, and $P[c_g|H_0]$ is its probability when BS is idle. Similarly, $P[c_m|H_1]$ and $P[c_m|H_0]$) are the corresponding probability for malicious node. $credit_1(g)$ can be got by the probability that local report matches the global decision minus the probability that the report conflicts the global decision (when it receives a penalty of one credit decrement). There are two cases where local report matches global decision: $(1-b)P_{11}$ stands for both local report and global decision are 1, and $b(1-P_{11})$ when both of them are 0. There are two cases for mismatching as well. $(1-b)(1-P_{11})$

for local being 1 while global being 0, and bP_{11} vice versa. So, there is:

$$\begin{aligned} P[c_g|H_1] &= (1-b)P_{11} + b(1-P_{11}) \\ &\quad - (1-b)(1-P_{11}) - bP_{11}, \\ &= (1-2b)(2P_{11}-1). \end{aligned} \tag{15}$$

Similarly, for $P[c_g|H_0]$ there is,

$$\begin{aligned} P[c_g|H_0] &= (1-a)(1-P_{10}) + aP_{10} \\ &\quad - (1-a)P_{10} - a(1-P_{10}), \\ &= (1-2a)(1-2P_{10}). \end{aligned} \tag{16}$$

For malicious node, in always-free attack, its local result is always 0, and it will receive one credit increment when the global decision is 0 and one credit penalty when 1. So we have:

$$\begin{aligned} P[c_m|H_1] &= b - (1-b) = 2b-1, \\ P[c_m|H_0] &= (1-a) - a = 1-2a. \end{aligned} \tag{17}$$

If we define the duty cycle of a BS as P_{use}, there is $P(H_1) = P_{use}$, $P(H_0) = 1 - P_{use}$. We denote $P[c_g]$ as the probability a good node gets credit increment per round, and $P[c_m]$ for malicious node. There is

$$\begin{aligned} P[c_g] &= P_{use}P[c_g|H_1] + (1-P_{use})P[c_g|H_0], \\ P[c_m] &= P_{use}P[c_m|H_1] + (1-P_{use})P[c_m|H_0]. \end{aligned} \tag{18}$$

We estimate the average probability of credit increment per round for all nodes as $P[c_{avg}] = P[c_g](n-m)/n + P[c_m]m/n$. And w_g and w_n are the expected weight after N_{round} rounds of test for good node and malicious node respectively. According to (3), there is

$$\begin{aligned} w_g &= \frac{P[c_g]N_{round}P_{neigh} + g}{P[c_{avg}]N_{round}P_{neigh} + g}, \\ w_n &= \frac{P[c_m]N_{round}P_{neigh} + g}{P[c_{avg}]N_{round}P_{neigh} + g} \end{aligned} \tag{19}$$

where P_{neigh} defines the probability of a close node to be the neighbor of the node of interest during the whole process, and is a function of the nodes' transmission range, their average moving speed and the sensing inteval. Roughly, $P_{neigh} = f(l_{range}, v_{avg}, t_{int}) \approx 1 - (0.5v_{max}t_{int})/l_{range}$.

EWSPRT introduces soft decision. The soft decision reaction function $f_2(u_i)$ works as a factor of the weight. The expectation of $f_2(u_i)$ is

$$E[f_2(u_i)] = 1 - \frac{1}{2}\int_{\gamma_0}^{\gamma_1} exp^{(-\frac{x^2}{2})}dx \tag{20}$$

EWSPRT also introduces the best of rest strategy, which can also be modeled as an equivalent factor f_{BoR}, which stands for the ratio of the applied weight in the test with best of rest to the weight in the test if only randomly polling is used, which is the case for WSPRT. The sorted sequence of weights can be modeled as a decending arithmatic progression sequence. We estimate the maximum credit among all nodes as $P[c_{max}] = (1 + P[c_g])/2$, therefore, the first item in the sorted weight sequence, which is also the maximum, is $w_1 = w_{max} = (P[c_{max}]N_{round}P_{neigh} + g)/(P[c_{avg}]N_{round}P_{neigh} + g)$. As mentioned, the mean of credits of all nodes, is c_{avg}. Since in EWSPRT, weight is normalized with average credit $P[c_{avg}]$, the mean of this weight sequence, should be the mean of credits normalized with the average credit. So there is $w_{avg} = (P[c_{avg}]N_{round}P_{neigh} + g)/(P[c_{avg}]N_{round}P_{neigh} + g) = 1$. With these assumptions, $f_{BoR} = (w_1 + w_{avg})/(2w_{avg}) = (w_{max} + 1)/2$. Its details are provided in Appendix I.

Combined with the best of rest strategy, truncated test emphasizes more on the system stability and plays less significant role in the sampling numbers. So its influence will be neglected in this model.

After we adopt news features into EWSPRT, the weights really used in the test should be modeled as

$$\begin{aligned} w_g &= f_{BoR}E[f_2(u_i)]\frac{P[c_g]N_{round}P_{neigh} + g}{P[c_{avg}]N_{round}P_{neigh} + g}, \\ w_m &= f_{BoR}E[f_2(u_i)]\frac{P[c_m]N_{round}P_{neigh} + g}{P[c_{avg}]N_{round}P_{neigh} + g}. \end{aligned} \tag{21}$$

B.2 Modeling of Test Module

$E[L(W)|H_1]$ and $E[L(W)|H_1]$ are derived the same way as SPRT in [12], there is

$$\begin{aligned} E[L(W)|H_1] &= (1-b)\log\lambda_1 + a\log\lambda_0, \\ E[L(W)|H_0] &= (1-a)\log\lambda_0 + b\log\lambda_1. \end{aligned} \tag{22}$$

However, for EWSPRT, the difference exists in the expectation of probability ratio because, with the existence of malicious nodes, more cases need to be considered. A benign node can report $u_i = 1$ (where P_{11}/P_{10} is used as probability ratio) as well as $u_i = 0$ (where P_{01}/P_{00} is used). On the other hand, malicious nodes always report 0 (P_{01}/P_{00} is used). Using (17), we have

$$\begin{aligned} E[L(PR)|H_0] &= \frac{n-m}{n}P_{10}w_g\log(\frac{P_{11}}{P_{10}}) \\ &\quad + [\frac{n-m}{n}(1-P_{10})w_g + \frac{m}{n}w_m]\log(\frac{P_{01}}{P_{00}}) \\ E[L(PR)|H_1] &= \frac{n-m}{n}P_{11}w_g\log(\frac{P_{11}}{P_{10}}) \\ &\quad + [\frac{n-m}{n}(1-P_{11})w_g + \frac{m}{n}w_m]\log(\frac{P_{01}}{P_{00}}). \end{aligned} \tag{23}$$

According to (11) and (12), the expectation of total sampling number is

$$\begin{aligned} E[Sam] &= E[Sam|H_1]P(H_1) + E[Sam|H_0]P(H_0), \\ &= \frac{E[L(W)|H_1]}{E[L(PR)|H_1]}P_{use} + \frac{E[L(W)|H_0]}{E[L(PR)|H_0]}(1-P_{use}) \end{aligned} \tag{24}$$

where $E[L(W)|H_1]$ and $E[L(W)|H_0]$, $E[L(PR)|H_1]$ and $E[L(PR)|H_0]$ are derived from (18) and (19), respectively.

N_{neigh} is defined as the average number of neighbors each node (includes itself) has within its transmission range. We define the transmission range of each node as l_{range}, then, the transmission area is $S_{tx} = \pi(l_{range})^2$. S_{area} denotes the area where n nodes evenly distribute, so there is:

$$N_{neigh} = \frac{S_{tx}}{S_{area}}n + 1 = \frac{\pi(l_{range})^2}{S_{area}}n + 1. \qquad (25)$$

Finally, using (21) and (22), the target matrix, the average sampling number per neighbor, should be

$$E_{avg}[Sam] = \frac{E[Sam]}{N_{neigh}}. \qquad (26)$$

For always-false and always-busy attacks, the differences lie in the expectation of credits in (13), (14) and the logarithmic expectation of probability ratio in (19).

For WSPRT, it is much easier to model since all new features like soft decisions will be eliminated.

Though EWSZOT itself is not a probability ratio test, if we let both sides of (10) be the index of 2, there is

$$2^W = 2^{\left(\sum_{i=0}^{m}(-1)^{u_i+1}\cdot w_i\right)} = \prod_{i=0}^{m}\left(2^{(-1)^{u_i+1}}\right)^{w_i}.$$
$$\begin{cases} 2^W \geq 2^q \Rightarrow \text{accept } H_1, \\ 2^W \leq 2^{-q} \Rightarrow \text{accept } H_0, \\ 2^{-q} < 2^W < 2^q \Rightarrow \text{take another round.} \end{cases} \qquad (27)$$

It is easy to find that when $u_i = 1$, $2^{(-1)^{u_i+1}} = 2$, and when $u_i = 0$, $2^{(-1)^{u_i+1}} = 2^{-1}$. So we can strictly regarded them as empirical and fixed value of probability ratio under hypothesis 1 and hypothesis 0 seperately. So we have

$$\frac{P[1|H_1]}{P[1|H_0]} = 2^1 \ \ (u_i = 1), \ \ \frac{P[0|H_1]}{P[0|H_0]} = 2^{-1} \ \ (u_i = 0). \qquad (28)$$

If we redefine $W' = 2^W$, and merge (26) into (25), there is

$$W' = \prod_{i=0}^{m}\left(\frac{P[u_i|H_1]}{P[u_i|H_0]}\right)^{w_i},$$
$$\begin{cases} \frac{P[1|H_1]}{P[1|H_0]} = 2^1, \frac{P[0|H_1]}{P[0|H_0]} = 2^{-1}, \\ W' \geq 2^q \Rightarrow \text{accept } H_1, \\ W' \leq 2^{-q} \Rightarrow \text{accept } H_0, \\ 2^{-q} < W' < 2^q \Rightarrow \text{take another round.} \end{cases} \qquad (29)$$

Now EWSZOT is also expressed in the form of probability test, it would be possible for us to use previously developed model to anticipate the sampling number of EWSZOT as well.

V. PERFORMANCE ANALYSIS

A. Simulation Network Model, Process, and Goal

In this section, we compare the performance of WSPRT, EWSPRT, and EWSZOT using simulation. Fig. 2 shows the simulation network model. In the simulation, the primary user, a TV tower, is located at $(D, 1000)$, working with a duty cycle of 0.2. A secondary ad-hoc network is composed of N secondary users, which are randomly distributed in a 2000 m × 2000 m square. The transmission range of every node is 250 m. There are N_a SSDF attackers among N secondary users. Three types of SSDF attacks mentioned before could be launched: Always-false, always-busy, and always-free. The random waypoint mobility model described in [13] is used to describe the movement of nodes, with a maximum speed of 10 m/s. HATA model specified in Section III.A.5 is used to describe the path loss effect. Thresholds are selected as $\gamma = -94$ dbm, $\gamma_1 = -89$ dbm, $\gamma_0 = -99$ dbm, $\lambda_1 = 10^{-6}$, $\lambda_0 = 10^{-6}$. New variables decribing the features of EWSPRT and EWSZOT are selected as $g = 5.51$, $Round_{max} = 100$, n_0 is uniformly distributed from -126 dbm to -86 dbm, and σ is uniformly distributed from 9.8 to 13.8.

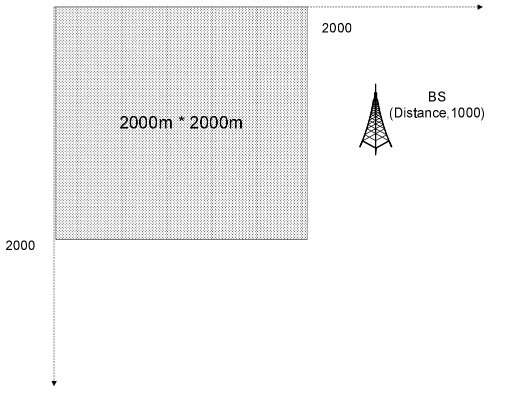

Fig. 2. Simulation network model.

The system executes the spectrum sensing every 20 second and the total simulation time is 1 hour. At each sensing time, the system records the reports from nodes, adjusts credits according to their correctness, and counts the events of false alarm, false negative, and the sampling number.

When simulation ends, four matrices (false positive rate, false negative rate, correct detection rate, and sampling numbers) are derived based on the statistics.

B. Simulation Result

We create various cases to get a complete view of the performance difference between WSPRT, EWSPRT, and EWSZOT.

B.1 Results Under Various Number of Malicious Nodes

First, we fix the total number of nodes $N = 100$, the distance of BS $D = 2500$, and the attack type could be all three types. The threshold needed in EWSZOT was set to $q = 15$, so that the error performance of EWSZOT would be comparable to other two. We observed the simulation goal under three attacks with various number of malicious nodes. Fig. 3 shows how three schemes work under always-busy attack. EWSZOT has the best false positive (FP) rate performance and the FP rate of EWSPRT is slightly less than that of WSPRT. Both EWSPRT and EWSZOT have a slightly larger false negative rate than WSPRT. EWSZOT can always hold the highest correct detection rate regardless of the increase of malicious node number, and EWSPRT always performs better than WSPRT for the correct rate.

Generally, EWSPRT and EWSZOT have a comparable error rate performance with WSPRT. The main improvement lies in the sampling numbers. Note that EWSPRT has almost one-

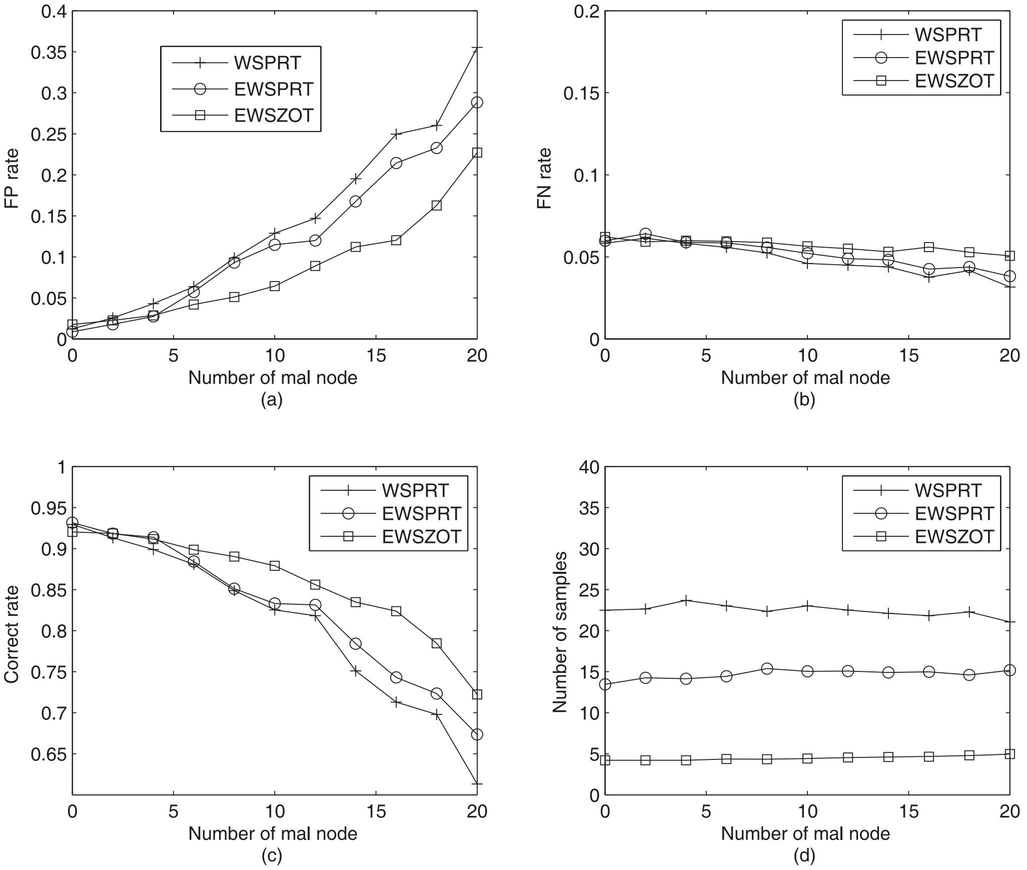

Fig. 3. Simulation results under various number of malicious nodes: (a) FP rate, (b) FN rate, (c) correct rate, and (d) sampling number.

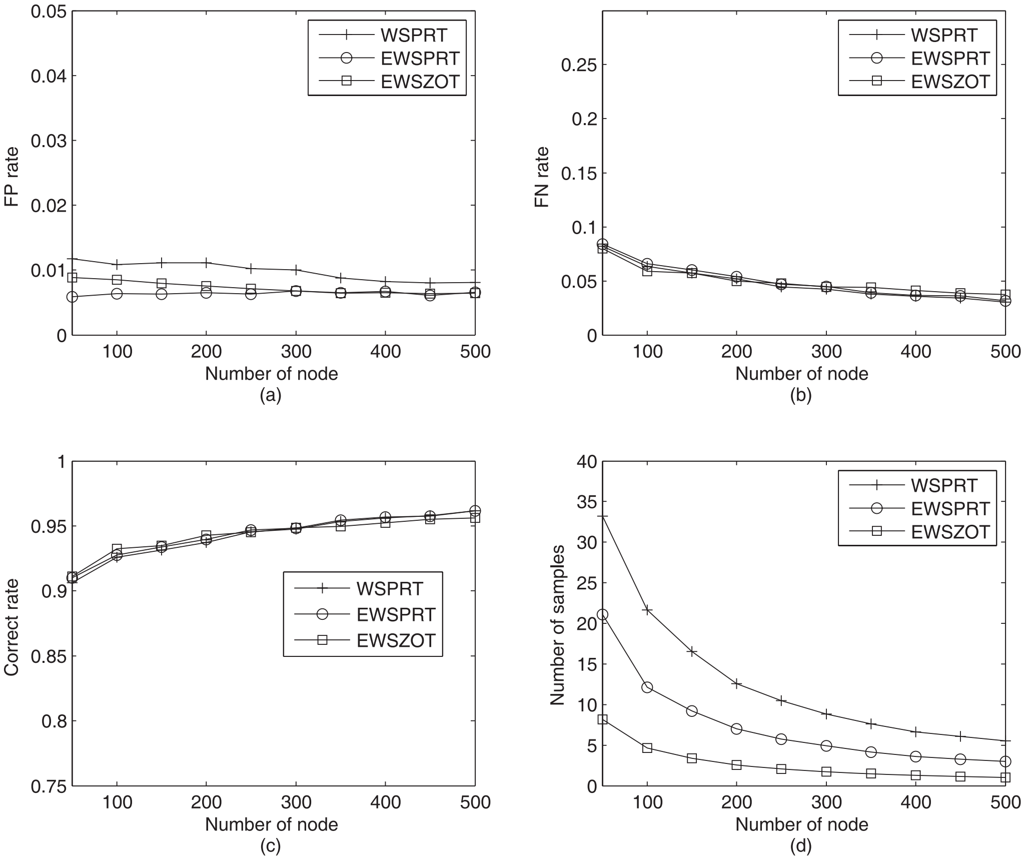

Fig. 4. Simulation results under various number of nodes: (a) FP rate, (b) FN rate, (c) correct rate, and (d) sampling number.

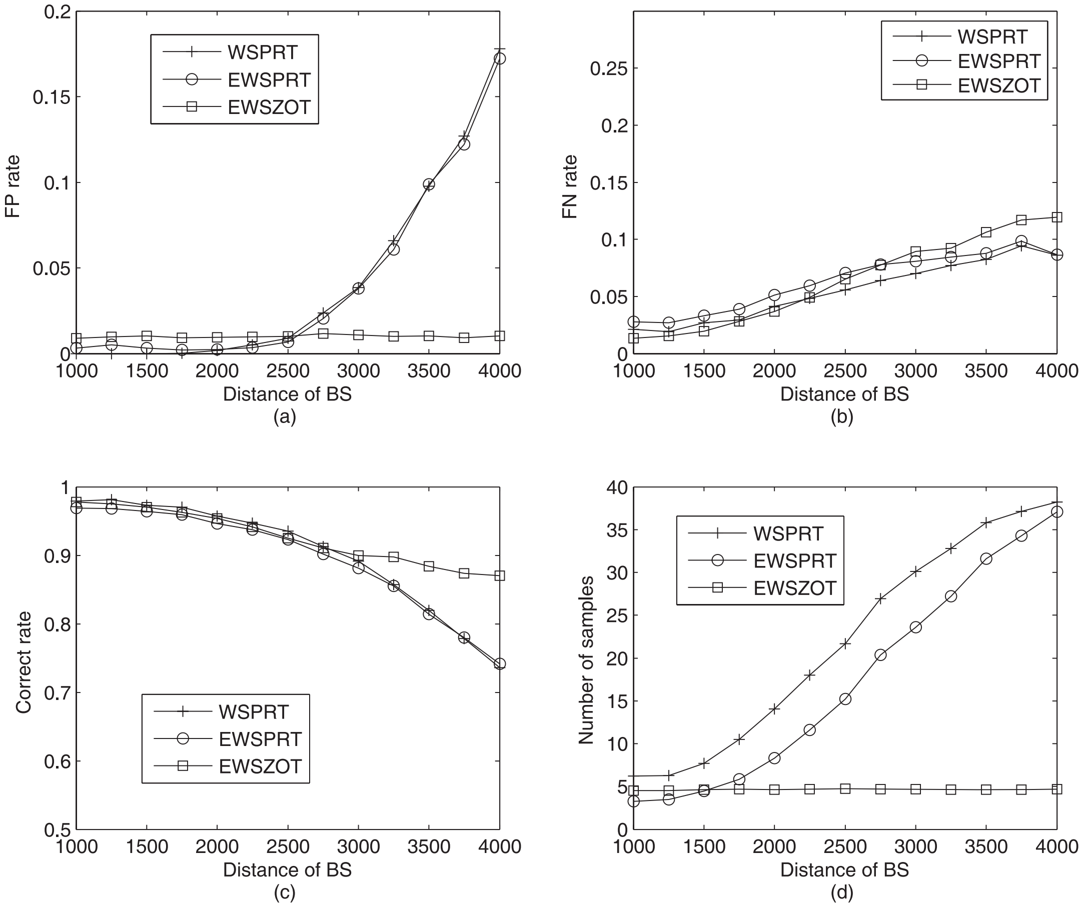

Fig. 5. Simulation results under distance of BS: (a) FP rate, (b) FN rate, (c) correct rate, and (d) sampling number.

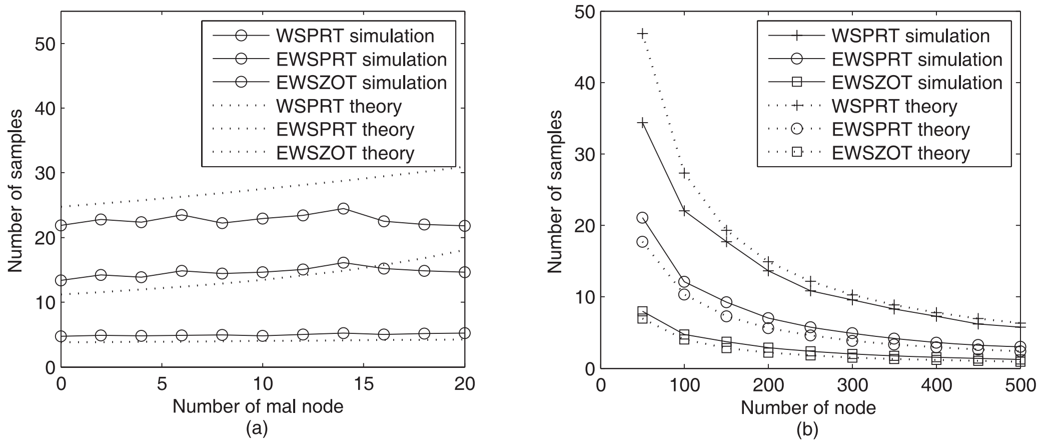

Fig. 6. Analysis vs. simulation: (a) Varying number of malicious nodes and (b) varying number of nodes.

half the sampling overhead of WSPRT, and EWSZOT has an even better one-fourth overhead. These are dramatic improvments since it is known from [10] that WSPRT has four times the overhead of SPRT, which means EWSPRT greatly mitigates the problem, and EWSZOT, with overhead comparable to the vulnerable SPRT, can address an SSDF attack perfectly without introducing any sampling overhead.

The simulation run under always-free attack and always-false attack achieved similar results: EWSPRT and EWSZOT reduce overhead by about 50% and 75%, respectively.

To simplify the analysis, in the latter part only one type of attack is used for comparison.

B.2 Results Under Various Number of Nodes

We set $D = 2500$ and $q = 15$ to see how the matrices would respond to a varying number of nodes N from 50 to 500 in an always-false attack. To be fair, the number of attackers N_a was always kept to 10% of the total node number.

Fig. 4 shows that for all the node number cases, EWSPRT and

EWSZOT always had comparable error rate performance and a much lower $1/2$ and $1/4$ sampling. This is consistent with the previous conclusion.

B.3 Results Under Distance of BS

If we set the node number $N = 100$, and the malicious nodes number $N_a = 10$, for always-false attack, we can observe how the target reacted to the change of BS distance. Fig. 5 shows that the error rates are comparable and sampling overhead decreased 50% and 75% separately. EWSZOT is the most robust scheme under longer distance; i.e. when a much weaker signal is received. It also can be seen that the performance dropped dramatically near 4000 m since from then on BS is out of the range of most nodes.

C. Comparison Between Analysis and Simulation Result

In this section, numerical results are compared with previous simulation data to check the validity of the established model. We use two cases as examples. Fig. 6(a) shows a comparison of the sampling number under varying number of malicious nodes when $N = 100$, $D = 2500$, and the attack type is always-free. Fig. 6(b) compares three schemes under varying number of nodes when $D = 2500$, the attack type is always-free, and malicious nodes are 10% of total nodes. From the figure, we note that the model matches the simulation results well. The model for WSPRT always anticipates the sampling number slightly larger than the simulation while model for EWSPRT and EWSZOT is slightly smaller. The model performs better with a higher node density, where nodes are distributed more evenly in the whole network. While only few nodes (e.g. $n = 50$) are scattered in large area, it is more difficult for each node to have enough neighbors. Since there is no protection for system stablity, extreme case is more likely to happen and it could be hard for the test to meet either bound. That is the reason the gap between analysis and simulation are greater with fewer nodes.

VI. CONCLUSION

In this paper, we discussed the reason most data fusion schemes in cooperative spectrum sensing are vulnerable to spectrum sensing data falsification (SSDF) attack which fabricates the sensing data towards the fusion center. Weighted sequential probability ratio test (WSPRT) was the only proposed scheme addressing this attack, although its sampling overhead is extremely large. Thus, we proposed two enhanced algorithms, enhanced weighted sequential probability ratio test (EWSPRT) and enhanced weighted sequential zero/one test (EWSZOT), which are robust against SSDF and achieve much better performance than WSPRT. Simulation results supported our expectation, and showed that sampling overhead can be reduced by 40% (EWSPRT) and 75% (EWSZOT), respectively. We also developed a theoretical analysis to model the sampling overhead, which matches the simulation results quite well.

APPENDICES

I. Derivation of f_{BoR}

We assume that each node has a different weight, and after sorting algorithm under best of rest strategy, they form a descending arithmetic progression sequence. The length of sequence is n. We describe it as $\{w_1, w_2, \cdots, w_n\}$, where w_1 is the largest weight, w_n is the smallest. And we define the mean of the sequence, w_{avg}. There is $w_avg = (w_1 + w_2 + ... + w_n)/n$.

The test could be finished after polling any element in the $\{w_1, w_2, \cdots, w_n\}$, Though not precise, we assume each case has an equal probability of $1/n$. Under each case, we'd like to get the ratio between the weight under best of rest stretegy and that under random polling. f_{BoR} is the expectation of the ratio under all cases.

For the random polling, though it is not clear which node is polled at a specific moment, statistically, the expection of the weight will converge to the mean w_{avg}.

So if the test ends after w_1, one node is polled, best of rest has a weight of w_1, random polling has a weight of w_{avg}, the weight ratio is

$$f_{BoR(1)} = \frac{w_1}{w_{avg}} = \frac{w_1 + w_1}{2w_{avg}}.$$

If the test ends after w_2, two nodes are polled, best of rest has a weight of $w_1 + w_2$, random polling has a weight of $2w_{avg}$, the ratio is

$$f_{BoR(2)} = \frac{w_1 + w_2}{2w_{avg}}.$$

Similarly,

$$\begin{aligned} f_{BoR(3)} &= \frac{w_1 + w_2 + w_3}{3w_{avg}} \\ &= \frac{w_1 + \frac{1}{2}(w_1 + w_3) + w_3}{3w_{avg}} = \frac{w_1 + w_3}{2w_{avg}} \end{aligned}$$

$$\begin{aligned} f_{BoR(4)} &= \frac{w_1 + w_2 + w_3 + w_4}{4w_{avg}} \\ &= \frac{w_1 + (w_1 + w_4) + w_4}{4w_{avg}} = \frac{w_1 + w_4}{2w_{avg}} \end{aligned}$$

$$\vdots$$

$$f_{BoR(n)} = \frac{w_1 + w_n}{2w_{avg}}.$$

So there is,

$$\begin{aligned} f_{BoR} &= \frac{1}{n} f_{BoR(1)} + \frac{1}{n} f_{BoR(2)} + ... + \frac{1}{n} f_{BoR(n)} \\ &= \frac{w_1 + \frac{1}{n}(w_1 + w_2 + ... + w_n)}{2w_{avg}} \\ &= \frac{w_1 + w_{avg}}{2w_{avg}}. \end{aligned}$$

REFERENCES

[1] FCC, "ET Docket No 03-222: Notice of proposed rule making and order," Dec. 2003.

[2] Q. H. Mahmoud, *Cognitive Networks Towards Self-Aware Networks*, John Wiley & Sons, 2007.

[3] C. Mathur and K. Subbalakshmi, "Digital signatures for centralized DSA networks," in *Proc. IEEE CCNC*, 2007.

[4] A. Sahai, N. Hoven and R. Tandra, "Some fundamental limits in cognitive radio," in *Proc. Alleton Conf. on Commun., Control and Computing*, Oct. 2004.

[5] H. Tang, "Some physical layer issues of wide-band cognitive radio system," in *Proc. IEEE DySPAN*, 2005.

[6] R. Chen and J.-M. Park, "Toward secure distributed spectrum sensing in cognitive radio networks," in *Proc. DSA workshop*, Sept. 2006.

[7] A. Pandharipande, J.-M. Kim, D. Mazzarese, and B. Ji, "IEEE P802.22 wireless RANs: Technology proposal package for IEEE 802.22." [Online]. Available: http://www.ieee802.org/22/, Nov. 2005.

[8] L. Lu, S.Y. Chang, J. Zhang, and L. Qian,"IT technology proposal clarifications for IEEE 802.22 WRAN systems." [Online]. Available: http://www.ieee802.org/22/, Mar 2006.

[9] B. K. Ghosh, *Sequential Tests of Statistical Hypotheses*, Addison-Wesley, 1970.

[10] R. Chen, J.-M. Park, and Kaigui Bian, "Robust distributed spectrum sensing in cognitive radio networks," in *Proc. IEEE INFOCOM*, 2008.

[11] S. M. Kay, *Fundamentals of Statistical Signal Processing: Detection Theory*, Prentice-Hall, 1998.

[12] P. K. Varshney, *Distributed Detection and Data Fusion*, Springer-Verlag New York, 1997.

[13] C. Bettstetter, G. Resta, and P. Santi, "The node distribution of the random waypoint mobility model for wireless ad hoc networks," *IEEE Trans. Mobile Comput.*, vol. 2, no. 3, July–Sept. 2003.

[14] T. S. Rappaport, *Wireless Communications: Principles and Practice*, Prentice Hall, 1996.

Feng Zhu was born in Shanghai, China, on March 10, 1984. He received the B.E. degree in Communication Engineering from Fudan University, Shanghai, China, and the M.E. degree in Electrical Engineering from the Seoul National University, Seoul, South Korea, in 2006 and 2008, respectively. From September 2009, he was in the DMC R&D Center, Samsung Electronics. His major interests are wireless network and security.

Seung-Woo Seo received the B.S. and M.S. degrees from Seoul National University, Seoul, Korea, both in electrical engineering and the Ph.D. degree in electrical and computer engineering from Pennsylvania State University, University Park in USA. He was on the Faculty of the Department of Computer Science and Engineering, Pennsylvania State University, and was a Member of the Research Staff in the Department of Electrical Engineering in Princeton University, Princeton, NJ. In 1996, he joined the Faculty of Seoul National University, where he is currently a Professor in the School of Electrical Engineering. He has served as a Chair or a Committee Member in various international conferences and workshops including Infocom, Globecom, PIMRC, VTC, MobiSec, Vitae, etc. He also served for five years as a Director of the Information Security Center in Seoul National University. His research areas include computer & network security, future mobile & wireless networks, and system optimization.