Anomaly Detection in Cognitive Radio Networks Exploiting Singular Spectrum Analysis

Qi Dong[1], Zekun Yang[1] Yu Chen[1], Xiaohua Li[1], and Kai Zeng[2]

[1] Dept. of Electrical & Computer Engineering, Binghamton University, Binghamton, NY 13902, USA
qdong3,zyang26,ychen,xli@binghamton.edu
[2] Volgenau School of Engineering, George Mason University, Fairfax, VA 22030
kzeng2@gmu.edu

Abstract. Cognitive radio networks (CRNs) is a promising technology that allows secondary users (SUs) extensively explore spectrum resource usage efficiency, while not introducing interference to licensed users. Due to the unregulated wireless network environment, CRNs are susceptible to various malicious entities. Thus, it is critical to detect anomalies in the first place. However, from the perspective of intrinsic features of CRNs, there is hardly in existence of an universal applicable anomaly detection scheme. Singular Spectrum Analysis (SSA) has been theoretically proven an optimal approach for accurate and quick detection of changes in the characteristics of a running (random) process. In addition, SSA is a model-free method and no parametric models have to be assumed for different types of anomalies, which makes it a universal anomaly detection scheme. In this paper, we introduce an adaptive parameter and component selection mechanism based on coherence for basic SSA method, upon which we built up a sliding window based anomaly detector in CRNs. Our experimental results indicate great accuracy of the SSA-based anomaly detector for multiple anomalies.

Keywords: Cognitive Radio Networks, Anomaly Detection, Singular Spectrum Analysis

1 Introduction

The rigid spectrum allocation scheme regulated by governmental agencies leads to great deficit on spectrum band resources utility. The emergence of new intelligent spectrum allocation/re-allocation scheme, especially cognitive radio network (CRN), is studied elaborately in the last decade, due to the ever-increasing wireless applications. CRNs allow secondary wireless devices (unlicensed users) access spectrum resources dynamically without introducing major interference to licensed primary users (PUs). Because of the great difficulty and high complexity in dynamic spectrum access (DSA), and many open issues on security deployment, CRN study still stays in tentative phase and needs all-round improvements despite a large quantity of research. A well-designed CRN aims to serve for two purposes: to maximize the usage of spare spectrum resource as well as to protect the incumbent primary system from secondary network interference [1].

Due to the uncertainties in PU behaviours and unavoidable interloper including many malicious entities, it is extremely hard to maintain a stable cognitive radio system. Existence of anomaly behaviours, which include traditional anomaly security threats and newly emerged CRN-specific security threats, such as jamming, primary user emulation (PUE) attacks, spectrum sensing data falsification (SSDF), common control channel jamming, selfish users, intruding nodes and more [2], make the situation worse.

In CRNs, "anomaly situation" can be introduced by various malicious activities as well as by versatile unpredictable PU activities, both of which can cause degradation on link quality of CRNs. Sometimes, it is necessary to detect those anomaly situations without introduce much overhead to cognitive entities. This task belongs to change-point detection, which concerns the design and analysis of procedures for on-the-go detection of possible changes in the characteristics of a running (random) process. Specifically, the process is assumed to be continuously monitored through sequentially made observations (e.g., measurements), whose behaviors should suggest the process may have statistically changed. The aim is to conclude so within the fewest observations possible, subject to a tolerable level of the risk of false detection [3], [4]. The time instance at which the state of the process changes is referred to as the change-point, which is not known in advance.

In this work, we exploit the Singular Spectrum Analysis (SSA) theory to solve the anomaly detection problem in CRNs. The key contribution lies in an adaptive parameter and component selection mechanism that enables the SSA cope with the complexity in anomaly detection in CRNs. For the convenience of discussion, we assume an ON/OFF Markov PU activity model and PUE attacks anomaly model.

The rest of this paper is organized as the following. Section 2 provides background and some most closely related work. Section 3 describes the system model with interference to SU activities. Section 4 explores a study of SSA on CRNs anomaly detection. Then Section 5 reports our experimental results. Section 6 concludes this paper.

2 Background and Related Work

To date, there are significant amount of works have been presented to address various CRN seciruty issues. For instance, the counter-measures to SSDF attacks include deploying reputation metric to denote the scale of trustworthy of each user [5], or reporting continuous sensing result to minimal attacks [6]. Game theory and Q-learning algorithm are often utilized to discuss attacker-SU action patterns [7] and against selfish users [8]. In confront of common control channel jamming attack, traditional communication technique of channel hopping is proved efficient for SUs exchanging channel information via multiple common control channels stochastically [9]. From the perspective of intrinsic features of CRNs, there is hardly in existence of an universal anomaly detection scheme.

SSA is introduced by Broomhead and King in 1986 [10]. Since then, it has shown its significant ability in a wide field of time series processing, such as finding data structure, extracting periodic pattern and complex trends, smoothing and change point detection [11],[12]. Wu *et al.* applied SSA for data preprocessing, associating with ANN, to predict daily rainfall-runoff transformation [13]. Oropeza and Sacchi presented multi-

channel singular spectrum analysis (MSSA) as a tool for simultaneously denoising and reconstructing seismic data [14]. Moskvina and Zhigljavsky developed an algorithm of change-point detection in time series, based on sequential application of SSA [15].

A PUE attack is that malicious entities mimic PU signals in order to either occupy spectrum resource selfishly or conduct Denial of Service (DoS) attacks. PUE attacks can be easily implemented in CRNs and introduce great overhead on cognitive radio communication and cause chaos in dynamic spectrum sensing. Many detection techniques are based on geometrical information of the PU transmitter. In [16], the authors claimed that in order to achieve a better attack result, attackers intend to adjust transmission power according to PU activities. By implementing a variable detection method that can measure the received power at SUs, it achieved a good detection result. However, it requires priori knowledge of distances among the nodes in the wireless environment. On the other hand, it is not accurate to identify PU signal by measuring the RSS. Not only because signal strength may vary by a large magnitude over small area, but also the attackers can constantly change the transmitter position and transmission power to disguise themselves. Thus, researchers have tried to eliminate noisy from received signals by constructing a sensor network [17], which introduces overhead to CRNs.

Comparing to all discussed PUE attack counter-measures, we do not aim at sifting adversaries' signal from PU signal. Instead, our method is able to detect abnormal activities without acquiring any priori information of PUs or malicious entities, and neither is there any overhead to PUs.

3 System Model

Let us consider a typical centralized CRN. Due to the opportunistic nature of cognitive radio spectrum access methods and intricate wireless channel traffic model, a method for detecting abnormal activities in CRNs is not always universally applicable to all situations. In an environment of non-deterministic PU traffic pattern, it is difficult to precisely predict channel idle periods. In a distributed CRN, all cognitive nodes share the spectrum resource with incumbent users, thus a single cognitive node can hardly be aware of an anomaly at system level. Therefore, a smart attacker can take advantage of the nature that channel idle period could fluctuate dramatically, and disguise as the PU to occupy spectrum resource selfishly. In our system model, an online detection technique is designed to fit many wireless environments regardless of channel fluctuation, by simply inspecting cognitive nodes' activities.

3.1 Assumptions

We made the following assumptions for our model:

- The CRN consists of several cognitive nodes that can dynamically access spectrum resource, and a fusion center (FC) that collects cognitive nodes' data flow information, and detects abnormal activates;
- The total available spectrum resource is composed by multiple PU channels. The channels are independent to each other on traffic pattern. The traffic model for each channel is non-deterministic with fluctuation in both busy and idle periods;

- The spectrum resource is intensively used by CRNs in which the cognitive node stores the sensed channel state at local and will transmit data when the channel is not used by other cognitive nodes, while a minimal transmission spectrum opportunity time slot T_{min_tx} is required;
- Every cognitive node will broadcast a packet P_{num} to FC after a transmission period T_{period} via an idle channel. P_{num} contains information of the number of received data packets in last T_{period}. The overhead to each cognitive node is minimal for only one extra packet is required for every T_{period};
- It is always feasible for each cognitive node to find an idle channel to broadcast P_{num}, because $T_{period} \gg T_{busy}$. (T_{period} is in the order of second, while T_{busy} is always in order of millisecond [18]);
- Attackers are smart enough to mimic PUs' signals, and they can conduct secret attack without introducing great fluctuation to the entire CRN;
- The FC can perform online anomaly detection based on the integrated statistics from all cognitive nodes.

3.2 Wireless Traffic Model and Analysis

Dynamic spectrum access (DSA) allows a cognitive radio to assign SUs some licensed bands temporarily, in an opportunistic and non-interfering manner. Therefore, the information about the spectrum occupancy pattern of the PUs is necessary. Because PU channels are independent with each other, it is feasible to analyze the model of an individual wireless channel. At a particular time point and a geographical location, a primary radio channel is either busy or idle. As illustrated in Fig. 1, T^i_{busy} denotes PU busy time slot on the i-th channel, which indicates there is PU activity exist in such channel. T^i_{idle} stands for PU idle time slot on i-th channel, which means no primary user is occupying this channel. Hence, the primary radio channel's spectrum occupancy pattern, which is also the PU's traffic pattern, describes the distribution of the durations of the busy and idle time slots.

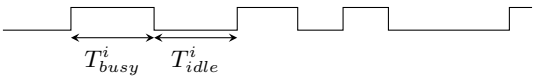

Fig. 1. Channel model of idle & busy time.

Basically, there are two classes of traffic patterns in wireless environment: 1) Deterministic patterns, where the duration of either idle time (T_{idle}) or busy time (T_{busy}), if not both, is fixed; and 2) Stochastic patterns, where the start time and duration of both states are random and be modeled with statistical properties [19]. For the former, the appearance of an attack will certainly change the deterministic patterns, making it easy to detect. For the later, which is more likely in real world situation, since the state exchanges randomly, it is hard to determine if the change of the occupancy pattern is related to an attack. Therefore, considering a continuous-time model, the channel remains in one state for a random period before switching to the other state. It is proven to be

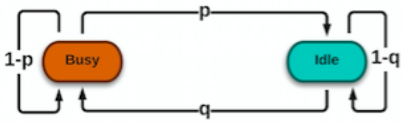

Fig. 2. PU channel ON/OFF two-state Markov Renewal Model.

efficient that the PU activity can be modeled as continuous-time, alternating ON/OFF two-state Markov Renewal Process (MRP) [20], as shown in Fig. 2.

In this model, both T^i_{idle} and T^i_{busy} can be regarded as independent and identically distributed (i.i.d.) process, where the PU activity arrival follows Poisson distribution. The continuous time span, i.e. T^i_{idle} and T^i_{busy}, follows exponential distribution, if PU activity arrival is a Poisson process [21].

The length of T^i_{idle} is critical for CRNs when exploiting the spectrum resource. A high time resolution of PU activity pattern may cause futile spectrum resource. For example, a successful packet transmission in CRNs requires a minimal transmission time span T_{min_tx}. In some intermittent PU activity channel, T^i_{idle} is usually too small for a complete transmission by SU, where $T^i_{idle} < T_{min\_tx}$. A valid idle transmission slot requires $T^i_{idle} > T_{min\_tx}$.

With a stable PU transmission pattern, there is a corresponding stable spectrum resource pool for CRNs. Any anomaly behaviour introduced either by malicious entities or PU itself, will cause variation on rate parameters λ_{idle} and λ_{busy}, and afterwards the available SU transmission time span T_{tx}. In addition, a certain external symptom will show at FC as a change of the integrated packet flow P_{num}.

In this paper, we implement anomaly detection based on the channel information P_{num} from all cognitive nodes, without knowing prior information λ_{idle} and λ_{busy}.

4 SSA-based Anomaly Detection

When the channel fluctuation is not considered, it is straightforward to attribute a deterioration of CRN channel quality to an anomaly in certain category. From the viewpoint of the FC, an anomaly results in the decrease of the overall receiving packets rate of all the SUs. Therefore, SSA algorithm is introduced to detect the change of the overall packets rate. The basic algorithm of SSA is described as following.

Assume $\mathbb{X} = (x_1, x_2, \cdots, x_N)$ is a real-value time series with the length of N. A sliding window, with a fixed window length of M, is adopted to truncate $\mathbb{X}$ and get a series of lagged vectors, and then, transform these vectors to a trajectory matrix X. The trajectory matrix X includes the whole information of the original time series $\mathbb{X}$. The columns X_j of the trajectory matrix X can be considered as vectors in an M-dimensional space $\mathbf{R}_M$. A particular combination of a certain number l of the Singular Value Decomposition (SVD) eigenvectors determines an l-dimensional subspace $\mathcal{L}_l$ in $\mathbf{R}_M$, $l < M$. The M-dimensional data $X_1, \cdots, X_K$ is then projected onto the subspace $\mathcal{L}_l$.

In our system model, FC gets the packets rate in every T_{period} from each single SU, and calculates the whole network's packets rate $\mathbb{X} = (x_1, x_2, \cdots, x_N)$, which is the object of the SSA processing. The SSA processing of $\mathbb{X}$ includes the following steps.

Step 1: Embedding

Map the vector $\mathbb{X}$ into an $M \times K$ matrix X,

$$X = \left[\overrightarrow{X_1}, \overrightarrow{X_2}, \cdots, \overrightarrow{X_K}\right] = (x_{i,j})_{i,j=1}^{M,K} \tag{1}$$

$$\overrightarrow{X_i} = (x_i, \cdots, x_{M+i-1})', i = 1, \cdots, K \tag{2}$$

where $K = N - M + 1$. The matrix X is called trajectory matrix and the vectors $\overrightarrow{X_i}$ are called lagged vectors. Note that X is a Hankel matrix, which has the equal elements on it's skew-diagonals $i + j = const$.

Step 2: Singular Value Decomposition

Apply SVD procedure on the trajectory matrix X and obtain M singular values $\sqrt{\lambda_1}, \sqrt{\lambda_2}, \cdots, \sqrt{\lambda_M}$ (in decreasing order) and the corresponding left singular vectors $U_1, U_2, \cdots, U_M$, and right singular vectors $V_1, V_2, \cdots, V_M$. The collection $(\sqrt{\lambda_i}, U_i, V_i)$, $i = 1, 2, \cdots, M$ is called the $i-$th eigentriple of the SVD. According to the standard SVD terminology, λ_i and U_i are the eigenvalues and eigenvectors of matrix $R = XX'$, respectively, while V_i are the eigenvectors of matrix $R' = X'X$, V_i also are called the principal components. The eigentriple satisfies $V_i = X'U_i/\sqrt{\lambda_i}$. Note that the rank of X is d, which is also the rank of R, then $\lambda_i = 0$, where $i > d$. So the trajectory matrix X will be:

$$X = X_1 + X_2 + \cdots + X_d \tag{3}$$

where $X_i = \sqrt{\lambda_i} U_i V_i'$ are rank-one biorthogonal matrices, $i = 1, \cdots, d$.

Step 3: Grouping

Select a subset indices I of $\{1, 2, \cdots, d\}$, with l elements

$$I = \{i_1, i_2, \cdots, i_l\} \tag{4}$$

such that

$$\bar{I} = \{1, 2, \cdots, d\}/I \tag{5}$$

Then the representation turns to

$$X = X_{i_1} + X_{i_2} + \cdots + X_{i_l} + X_{\bar{I}} \tag{6}$$

where $X_{\bar{I}} = \sum_{i \notin I} X_i$.

Step 4: Diagonal Averaging

Diagonal Averaging is used to transfer matrix $X_I = \sum_{i \in I} X_i$ into a time series (reconstruction). According to mathematic deduction, it is the component-sum of the original series $\mathbb{X}$.

$$x_i = \begin{cases} \frac{1}{i}\sum_{j=1}^{i} x_{j,i-j+1} & for\ 1 \le i < M \\ \frac{1}{M}\sum_{j=1}^{M} x_{j,i-j+1} & for\ M \le i \le K \\ \frac{1}{N-i+1}\sum_{j=i-K+1}^{N-K+1} x_{j,i-j+1} & for\ K < i \le N \end{cases} \quad (7)$$

That is, the reconstructed series element x_i equals to the average of the corresponding matrix elements sharing the same location with that x_i appears in the trajectory matrix X. The main purpose of SSA is to decompose the original time series into several additive components, based on the assumption that this series is a sum of several simpler series. Specifically, a general descriptive model of the series that we use in SSA methodology is an additive model in which the components are trends, oscillations and noise. Fig. 3 shows a SSA decomposition of a section of CRN packet rate flow. The conception of "trend" depicts a components that is (i) not stationary and (ii) 'slowly varies' during the whole period of time that the series is being observed, as the first and second components shown in Fig. 3. Meanwhile, the 'oscillation' components can be divided into periodic and quasi-periodic, like the third and forth components shown in Fig. 3. Compared to them, the 'noise' component doesn't have a certain boundary with others. But generally speaking, 'noise' are aperiodic series and contribute less to the original series than others, like the fifth component.

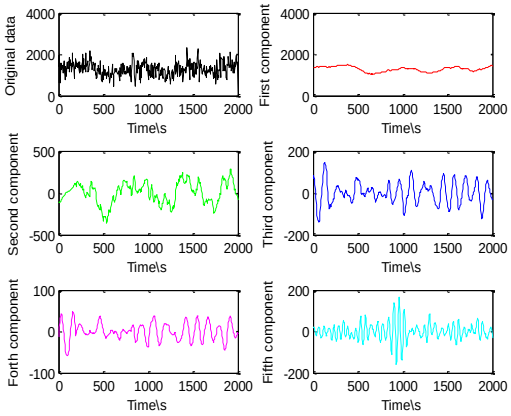

Fig. 3. SSA decomposition example.

Fig. 4 shows the differences among the three kinds of components in frequency domain. In general, the trend converge its energy at the low frequency region, shown by the first and second components. The oscillation corresponds to a peak at particular frequency, and always occur in pairs, shown by the forth and fifth component. Separating the whole series into these components and analyzing the LRRs for interpretable components are helpful to obtain reliable and meaningful SSA results. In our case, only the general variation trend of channel traffic flow are considered, so that the gentle periodical fluctuation and random channel error are ignored.

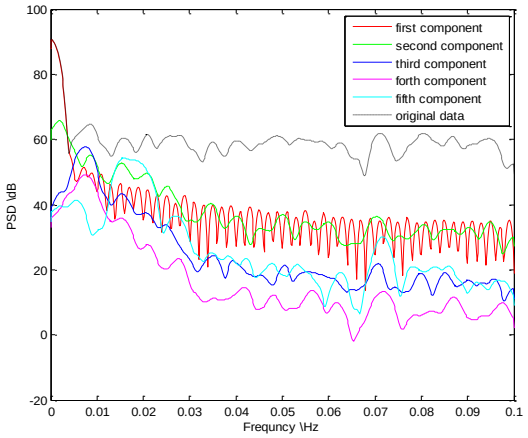

Fig. 4. Frequency domain analysis example.

Table 1. Network Setting

Simulation time	10000 $seconds$
Number of PUs	5
Number of SUs	10
Number of tranmission channles	5
SU spectrum sensing duration	10 ms
Transmission cycle of PUs	40 ms to 50 ms $(not\ fixed)$
Idle cycle of PUs	50 ms to 70 ms $(not\ fixed)$
Data packet size	2000 $bytes$
SU data channel rate	1.00 $Mbps$
SU transmission period T_{period}	5 $second$
Attacker hopping	Yes

5 Experimental Results

This paper reports a preliminary study, where the SSA method is employed to off-line anomaly detection problem. It illustrates the capability of SSA on capturing the operating trend of random packet rate.

5.1 Experimental Setup

Network Setting: A CRN is constructed using OMNET++ 4.6. In our implementation, SUs are randomly distributed in the environment and they can access to all channels in an opportunistic spectrum sensing manner. Each channel is legally allocated to a PU, whose duty cycle and idle cycle are stochastically distributed in a range. All SUs know neither geometry information of PUs, nor their broadcasting patterns. We test our detection method via introducing PUE attacks in experiment scenarios.

Table 2. Off-line Detector Setting

SSA Setting	Widow length	2000 $samples$
	Trajectory Matrix size M	300 $samples$
	Grouping element I	$\{1, 2\}$
Differentiation Setting	Interval	4 $samples$
Detector Setting	Threshold	15.12

Scenario Setting: This anomaly detector is applied to a PUE attack specified environment as the following. A malicious party is a smart attacker can start PUE attack at a randomly chosen time, and stop after accomplished sufficient attacks. For disguise purpose, the attacker hops among all spectrum channels randomly and implement secret PUE attacks. In this simulation, we applied 20 different seeds to construct 20 different CRN scenarios for anomaly detection. Among all scenarios, the attacker conduct PUE attack at 1500s, 5000s, 8000s respectively, and end at 3000s, 7000s, 9300s respectively. Specific parameter setting can be referred to table 1.

Detector Setting: An efficient detector requires a sufficiently long sliding window N in order to catch the principal system features, and the row size of trajectory matrix M should be restricted to $M < N/2$. In this off-line detection, since a long window is preferred, N can be pre-set as large as 2000 samples, and M can be pre-set as 300 samples. The detection threshold setting is critical in our detector. It can be generated by history observation data in normal non-attacking scenarios with proper guarding pad. Table 2 presents all detector parameters.

5.2 Anomaly Detection Experimental Results

We present detection results with and without PUE attack respectively. In Fig. 5, the left-top subfigure shows an original packet flow synthesized by FC, which consists of oscillations and spikes. The right-top subfigure shows the data sequence reconstructed by the SSA process. Based on the features carried by principal components of the original data, the first and the second principal components, which are shown by the two subfigures in the bottom of Fig. 5, are selected for reconstruction due to the great oscillations in other components. The reconstructed data sequence shows clear changes around the time points: $1500s$, $3000s$, $5000s$, $7000s$, $8000s$, and $9300s$, when the PUE attacks started or ended. In contrast, Fig. 6 shows an original packet flow without PUE attack and the reconstructed data sequence from SSA. The reconstructed data indicates a smooth and steady CRN behavior. Besides the reconstruction process, our detection model contains a differential detector. Fig. 7 and 8 show the outputs of the differential detector corresponding to cases with and without PUE attack respectively. While the detector output in Fig. 7 also shows spikes at those six anomaly points, the output in Fig. 8 presents only small fluctuations. Threshold configuration in differential detector is critical. Thus, a training session is required to find threshold properly.

Based on our detection method, with respective history data training, our SSA based detection method achieved a detection rate of 84%, with false alarm rate of 8%, as shown in Table 3. However, we suffered an average delay of $45.96s$, which may be

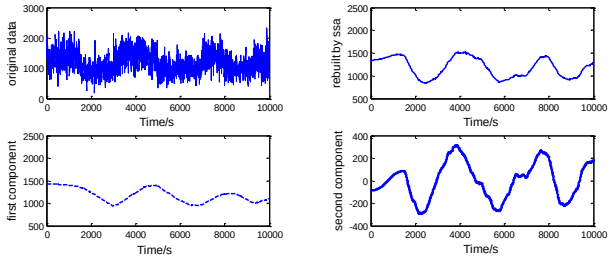

Fig. 5. Original data flow, SSA reconstructed data flow, and the first and the second principal components with attack.

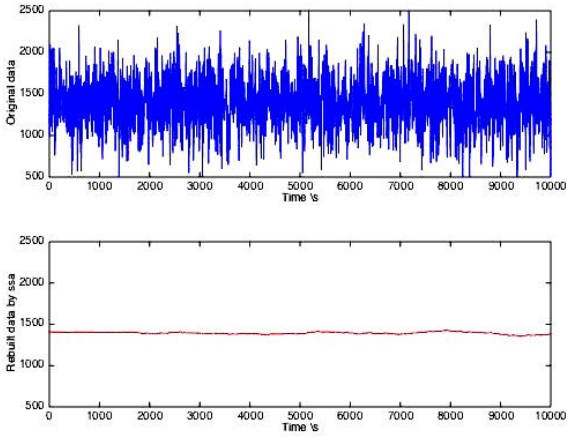

Fig. 6. Original data flow and SSA reconstructed data flow without attack.

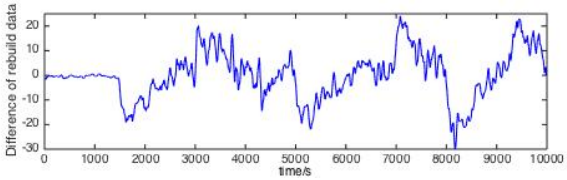

Fig. 7. Differential detector result with attack.

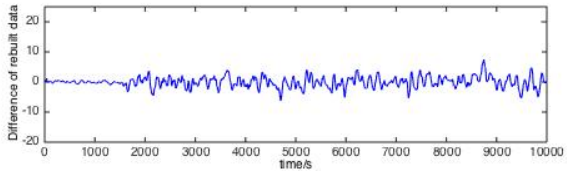

Fig. 8. Differential detector result without attack.

Table 3. Simulation Study Result

Detection Rate	84%
False Alarm	8%
Average Delay	$45.96s$

resulted from two factors. The results show the applicability of SSA based anomaly detection in CRN.

6 Discussions & Conclusions

From the experiment result, our SSA based anomaly detection method show high detection rate and low false alarm rate, even when encountered with subtle PUE attack. Although the simulation environment only considered two anomalies: PUE attack and PU abnormality, our proposed method can be used for many other anomaly detections, such as spectrum sensing data falsification (SSDF), jamming, because those anomaly activities will inevitably deteriorate communication condition of CRNs. However, our detection method suffers from relatively high detection delay. The delay is generally caused by two factors: 1) our light-weighted anomaly detector requires SUs report their transmission quality every several seconds, this relative low report time resolution will bring small amount of detection delay; 2) our detection method is built upon transportation-layer-based data flow, which is not very sensitive to physical layer anomaly activities (most common in CRNs).

In comparison to traditional attack-specified security insurance techniques, our SSA based anomaly detection method is applicable to most anomaly environment in CRNs. However, this method is no more than a detector. Due to its lightweight and ease of use, this method can be used as a rudimentary CRN network monitor and anomaly detector; once anomaly is detected, further action is needed to eliminate the anomaly.

Singular Spectrum Analysis (SSA) method possesses attractive features such as parametric model free that enables it to detect different categories of anomalies. The SSA algorithm is suitable for off-line batch data processing. Our experimental results verified the effectiveness of the SSA detector. While this algorithm can detect subtle changes with high accuracy, it suffers relatively long delays.

As our on-going effort, we are exploring to extend SSA to handle online anomaly detection problem. An adaptive parameter and component selection mechanism is being introduced to enable real time operation. In addition, we will also focus on the delay issue and aiming at an efficient anomaly detection scheme with both satisfactory accuracy and acceptable delays.

References

[1] Adelantado F, Verikoukis C (2013) Detection of malicious users in cognitive radio ad hoc networks: A non-parametric statistical approach. Ad Hoc Networks 11(8):2367–2380

[2] Esch J (2012) A survey of security challenges in cognitive radio networks: Solutions and future research directions. Proceedings of the IEEE 12(100):3170–3171
[3] Basseville M, Nikiforov IV, et al (1993) Detection of abrupt changes: theory and application, vol 104. Prentice Hall Englewood Cliffs
[4] Poor HV, Hadjiliadis O (2009) Quickest detection, vol 40. Cambridge University Press Cambridge
[5] Rawat AS, Anand P, Chen H, Varshney PK (2010) Countering byzantine attacks in cognitive radio networks. IEEE
[6] Min AW, Shin KG, Hu X (2009) Attack-tolerant distributed sensing for dynamic spectrum access networks. IEEE
[7] Wang B, Wu Y, Liu KR, Clancy TC (2011) An anti-jamming stochastic game for cognitive radio networks. IEEE Journal on Selected Areas in Communications 29(4):877–889
[8] Attar A, Nakhai MR, Aghvami AH (2009) Cognitive radio game for secondary spectrum access problem. IEEE Transactions on Wireless communications 8(4):2121–2131
[9] Cormio C, Chowdhury KR (2010) Common control channel design for cognitive radio wireless ad hoc networks using adaptive frequency hopping. Ad Hoc Networks 8(4):430–438
[10] Broomhead DS, King GP (1986) Extracting qualitative dynamics from experimental data. Physica D: Nonlinear Phenomena 20(2-3):217–236
[11] Rukhin AL (2002) Analysis of time series structure ssa and related techniques. Technometrics 44(3):290–290
[12] Yang Z, Zhou N, Polunchenko A, Chen Y (2015) Quick online detection of start time of disturbance in power grid. In: the IEEE GlobeCom 2015, Selected Areas in Communications Symposium: Smart Grid Communications Track, IEEE, pp 1–6
[13] Wu C, Chau K (2011) Rainfall–runoff modeling using artificial neural network coupled with singular spectrum analysis. Journal of Hydrology 399(3):394–409
[14] Oropeza V, Sacchi M (2011) Simultaneous seismic data denoising and reconstruction via multichannel singular spectrum analysis. Geophysics 76(3):V25–V32
[15] Moskvina V, Zhigljavsky A (2003) An algorithm based on singular spectrum analysis for change-point detection. Communications in Statistics-Simulation and Computation 32(2):319–352
[16] Chen Z, Cooklev T, Chen C, Pomalaza-Ráez C (2009) Modeling primary user emulation attacks and defenses in cognitive radio networks. IEEE
[17] Chen R, Park JM, Reed JH (2008) Defense against primary user emulation attacks in cognitive radio networks. Selected Areas in Communications, IEEE Journal on 26(1):25–37
[18] Mahamuni S, Mishra V (2014) Performance evaluation of spectrum detection in cognitive radio network. Int'l J of Communications, Network and System Sciences 7(11):485
[19] Marko Hoyhtya SP, Mammela A (2011) Improving the performance of cognitive radios through classification, learning, and predictive channel selection. In: ADVANCES IN ELECTRONICS AND TELECOMMUNICATIONS

[20] Rehmani MH, Viana AC, Khalife H, Fdida S (2013) Surf: A distributed channel selection strategy for data dissemination in multi-hop cognitive radio networks. Computer Communications 36(10):1172–1185

[21] Sriram K, Whitt W (1986) Characterizing superposition arrival processes in packet multiplexers for voice and data. IEEE journal on selected areas in communications 4(6):833–846