Cooperative Robust Sequential Detection Algorithms for Spectrum Sensing in Cognitive Radio

ArunKumar Jayaprakasam and Vinod Sharma,
Electrical Communication Engineering Department,
Indian Institute of Science, Bangalore 560012, India
Email: arunkumar.jayaprakash@yahoo.in, vinod@ece.iisc.ernet.in

Abstract— **We consider the problem of Spectrum Sensing in Cognitive Radio Networks. In our previous work we have developed DualCUSUM, a distributed algorithm for change detection and used it for cooperative spectrum sensing. The algorithm is based on sequential change detection techniques which optimally use the past observations. But DualCUSUM requires the knowledge of the channel gains for each of the secondary users. In this work we modify DualCUSUM to develop GLR-CUSUM which can work with imprecise estimates of the channel gains. Next we extend the algorithm to the case where the receiver noise power is also not known exactly. We show that the SNR wall problem encountered in this scenario is not experienced by our algorithm. We also analyze these algorithms theoretically.**

Keywords-Cognitive Radio, Cooperative Spectrum Sensing, Decentralized Sequential Detection, Robust Detection.

I. INTRODUCTION

Cognitive Radios are defined as radio systems which perform radio environment analysis, identify the spectral holes and then operate in those holes ([6], [10], [18]). In cognitive radio terminology *Primary user* refers to a user who is allocated the rights to use the spectrum. *Secondary user* refers to the users who try to use the frequency bands allocated to the primary user when the primary user is not using it.

Spectrum Sensing, an essential component of the Cognitive Radio technology involves, 1) identifying spectrum holes (white space) and 2) when an identified spectrum hole is being used by the secondary users, to quickly detect the onset of primary transmission. This needs to be done such that the guaranteed interference levels to the primary are maintained and there is efficient use of spectrum by the secondary. This involves detecting reliably, quickly and robustly, possibly weak, primary user signals. For example the IEEE 802.22 standard ([6]) requires a sensitivity of −116dBm.

For cognitive radios two typical scenarios have been identified. One is to use the white space in the TV broadcast channels ([10], [23]). In this case one often knows the timings when different TV broadcast channels are ON. A cognitive radio can use the channels which are free. During the time of broadcast at a channel also, a TV band can be used by cognitive users at locations where its transmission is weak, i.e., it cannot be used by a TV receiver at that place. For this a cognitive radio has to detect a spectral hole in space. If the cognitive radio is mobile then this can lead to detection of spectral hole in time also. In another scenario several IEEE 802.11 and Bluetooth devices may be sharing the ISM band (2.5 GHz). A device may use a channel when others are not using. This leads to detecting a white space in time where the channel may be available for use for a few msecs or secs ([9]).

The channel from the primary transmitter to a secondary user can be bad because of shadowing and time varying multipath fading. To alleviate this problem, Cooperative Spectrum Sensing ([5], [8], [13], [28]) is envisaged, whereby the spatial diversity inherent in radio environment is leveraged by allowing multiple secondary users to cooperate. This reduces the average time to detect the primary user. This in turn lowers the interference to the primary user (when it switches ON), while increasing the spectrum usage of the secondary (when the primary switches OFF).

Another important problem in spectrum sensing is the impact of modeling uncertainties, e.g., the noise distribution, noise power and/or channel gain may not be exactly known ([23]). Because cognitive radios have to detect primary signals at very low SNR, these modeling uncertainties can cause an SNR wall, a level below which spectrum sensing fails to be robust to modeling uncertainties. More recently it is shown in [17] that if the primary user goes ON and OFF at different times one may be able to detect this change in a reliable and robust manner and then SNR wall can be breached. Of course in this case one is looking for spectral holes in time.

In this paper we provide a spectrum sensing algorithm (for detecting spectral holes in time) which can be used cooperatively by multiple cognitive users distributed in space. It is robust to the channel gain and noise power uncertainties.

In the next section we describe briefly different Spectrum Sensing algorithms available in the literature.

A. Literature Survey

For spectrum sensing, primarily three signal processing techniques ([5]) are proposed in literature: Matched filter [20], Energy Detection [20] and cyclo-stationary feature detection [29]. Matched filtering is optimal. However it requires detailed knowledge of primary signaling. But when no knowledge about the primary user signal is assumed, an Energy Detector is optimal ([20]). Hence most of the literature is based on energy detection. Also IEEE 802.22 will provide for spectrum sensing via energy detection.

Cooperative spectrum sensing where the decisions of different secondaries are fused to obtain the final decision has been studied in [8], [19], and [28]. The problem of SNR Wall is introduced in

9781-4244-3941-6/09/$25.00 ©2009 IEEE

[20]. In [23] the existence of an SNR wall for an Energy Detector and the Matched Filter is presented.

Almost all of the above studies detect the spectrum hole from a single snapshot of observations from one or more nodes. In [22] we have applied a decentralized cooperative algorithm, DualCUSUM, developed in [1], for spectrum sensing. It is shown that it performs better than many other spectrum sensing algorithms developed recently. This algorithm is based on Sequential Detection techniques, in particular on CUSUM [16]. DualCUSUM can be implemented online, is iterative in nature and requires minimal computations at each step.

More recently sequential detection techniques have been used in [11] and [14] also. The algorithms in [11] and [14] are designed for a single node (thus have no cooperative features). The algorithm in [14] is tested for detecting a sinusoidal signal in white noise. [11] studies various sequential change detection algorithms when there is uncertainty in parameters and uses rank statistics.

B. Our Contribution

The algorithm in [22] requires the exact knowledge of the channel gains and noise power for each of the secondary user. In this paper we extend the algorithm in [22] to GLR-CUSUM which does not assume the precise knowledge of channel gains and noise power. In this algorithm instead of CUSUM, GLR ([12]) is used at the local nodes which also has some optimality properties. As a result our algorithm performs better than the algorithms in [11] and [17]. This comparison is reported elsewhere.

The paper is organized as follows. Section II describes our model. In Section III we start by briefly presenting the DualCUSUM algorithm. Then we present the GLR-CUSUM algorithm whereby the knowledge of the primary signal power is not required at the secondary nodes. We further modify the algorithm such that it can be applied under uncertainty in the mean noise power. We conclude this section by comparing the performance of the various algorithms. In Section IV we analyze these algorithms. Section V concludes the paper.

II. MODEL

Consider a Cognitive Radio System with L secondary users who sense a channel via Energy Detectors. The observations made on the channel by these users are processed and sent to a fusion center which makes a decision whether the channel is free or not. Then that decision is sent to all secondary users for possible use of channel.

The secondary nodes have to detect the change in the status of the channel in two situations. First, when the primary has been using the channel and it stops transmission. A short detection delay here will allow increased spectrum utilization opportunities for the secondary user. The second situation is when the channel has been free and the secondaries are using the channel. They need to sense the channel to see if the primary starts transmission. This is possible under the assumption that in between their transmissions, secondaries stop intermittently and sense the channel to see if the primary has started transmission. If yes the secondaries need to stop using that channel. A small detection delay here will cause minimal interference to the primary users. This setup can be used in the in-band spectrum sensing in IEEE 802.22.

We present our algorithm in the setup of the second scenario {primary OFF→ON} but the same algorithm works under the first scenario as well. In Section III we will provide details and also an example.

We consider a slotted system. During the OFF period of the primary, in the beginning of each slot, the secondaries only sense the channel and make observations. Based on those observations they transmit some data to the fusion center for making the final decision. If the fusion center decides that the channel is free, then the secondaries will use the rest of the slot for data transmission.

Let at random time T the primary starts transmission. Then in the k^{th} slot the signal received by the l^{th} secondary is,

$$X_{k,l} = N_{k,l}, \qquad k = 1, 2, ..., T-1,$$
$$X_{k,l} = h_l S_k + N_{k,l}, \quad k = T, T+1, ...$$

where h_l is the channel gain of the l^{th} user, S_k is the primary user signal and $N_{k,l}$ is observation noise at the l^{th} user. It is assumed that the fading is constant during the interval of the observation (say approximately for a duration of ON/OFF period). Slow fading scenarios with primary staying ON and OFF for a few seconds will approximately satisfy this. This assumption is commonly made in literature ([11], [13]). We also assume that $\{S_k, k \geq 1\}$ and $\{N_{k,l}, k \geq 1\}$ are independent and identically distributed (i.i.d) sequences independent of each other and T.

There are two commonly made assumptions about the pre and post change distributions of $X_{k,l}$. Some studies ([19], [24]) assume the pre-change distribution as $N(\mu_0, \sigma_l^2)$ and post-change as $N(\mu_0 + P_l, \sigma_l^2)$, i.e., the presence of primary is modeled as a change in mean of Gaussian noise, where $N(a,b)$ denotes Gaussian distribution with mean a and variance b. In the other model post change distribution is $N(\mu_0, \sigma_l^2 + P_l)$, i.e., presence of primary is modeled as increase in variance ([20]).

In this paper we provide algorithms for the mean change model. This model is relevant when the noise and interference are assumed to be log normally distributed, the logarithm of which takes Gaussian distribution ([26]). Here μ_0 denotes the mean noise power and σ_l^2 is the uncertainty in the noise power ([26], [28]). P_l denotes transmit power of the primary. Another scenario where this is a relevant model is when $X_{k,l}$ is the sum of energy observations at the secondary of sufficiently large samples ([19]). Although we present and analyze our algorithms in this setup, our algorithms are general and can be applied to other models as well.

The aim is to detect the change (at random time T) at the fusion center as soon as possible at a time τ ($\geq T$) using the messages transmitted from the L sensors with an upper bound on probability of false alarm (see below for precise statement). For this each of the L nodes uses its observation $X_{k,l}$ to generate a signal $Y_{k,l}$ and transmits to the Fusion Center. The data received at the fusion center is corrupted by the i.i.d. receiver noise Z_k at the fusion center. The fusion center uses the observations $Y_{k,1}, Y_{k,2}, \ldots, Y_{k,L}$ to decide between the two hypotheses H_0 (the primary is not transmitting) and H_1 (the primary is transmitting). If H_0 is chosen the secondaries continue to use the channel in slot k and the spectrum sensing session continues. If H_1 is detected, the secondaries typically switch over to an alternate channel. To transmit $Y_{k,1}, Y_{k,2}, \ldots, Y_{k,L}$ from the L secondaries to their fusion node, they need a Multiple Access Channel (MAC) protocol. Time Division Multiple Access (TDMA) is the most commonly used protocol. However we will allow all users to transmit simultaneously and use physical layer fusion (see step 3 of the algorithm in Section III.A). This reduces transmission delays.

We develop a robust cooperative algorithm for spectrum sensing in this setup. Let $P_{FA} = P(\tau < T)$ be the probability that the fusion center decides that the primary has started transmitting while it has not. We state the goal of the problem as:

$$\min EDD \triangleq E[(\tau - T)^+], \quad subject\ to\ P_{FA} = P(\tau < T) \le \alpha \tag{1}$$

where EDD denotes the Expected Detection delay, i.e., the average number of slots needed to detect the presence / absence of the primary. α is the FAR (False alarm Rate) constraint and $x^+ = \max(x,0)$. Thus we are seeking a scheme which minimizes the average detection under a FAR constraint. This is a very important requirement for cognitive radio.

In the distributed setting there is no known optimal solution of (1) (see [4], [25] and [27] for a state-of-the-art survey). However the algorithm in [22] uses many desirable features to provide a performance better than any algorithm known to us so far.

III. COOPERATIVE SENSING ALGORITHMS

We present DualCUSUM which was developed in [1]. Then we will generalize it.

DualCUSUM uses CUSUM algorithm at the secondary nodes as well as the fusion node. The first two steps in the DualCUSUM explained below present the CUSUM algorithm used at the secondary nodes. A secondary node transmits only when its CUSUM crosses threshold γ. The fusion node gets the input from all nodes and uses another CUSUM to make the final decision on whether a primary is present or not.

A. DualCUSUM Algorithm

1) Each of the secondary users l runs Parametric CUSUM algorithm ([16]):

$$W_{k,l} = \max(0, W_{k-1,l} + \xi_{k,l}), \; W_{0,l} = 0 \quad where,\; \xi_{k,l} = \log[f_{1,l}(X_{k,l}) / f_0(X_{k,l})], \tag{2}$$

$f_{1,l}$ is the density of $X_{k,l}$ under H_1 and f_0 is the density of $X_{k,l}$ under H_0.

2) Secondary user l transmits b at time k (i.e., declares a change locally), only if $W_{k,l} > \gamma$: $Y_{k,l} = b1_{\{W_{k,l} > \gamma\}}$. The parameters b and γ are chosen appropriately. This step allows to save energy and cause less interference to others.

3) At the fusion center we assume physical layer fusion: $Y_k = \sum_l Y_{k,l} + Z_k$ where Z_k is i.i.d. noise at the fusion node, i.e., at time k the fusion node receives Y_k when $Y_{k,l}$ has been transmitted by node l.

4) Change detection at the fusion center via CUSUM: The fusion node runs CUSUM based on observations Y_k: $F_k = \max\{0, F_{k-1} + \log \frac{g_I(Y_k)}{g_0(Y_k)}\}$ where g_0 is the density of Z_k and g_I is the density of $Z_k + bI$, I being a design parameter.

5) The Fusion Center declares a change at time $\tau(\beta, \gamma, b, I)$ when F_k crosses a threshold β: $\tau(\beta, \gamma, b, I) = \inf\{k : F_k > \beta\}$.

Although DualCUSUM is not probably optimal, it has some desirable features: (i) it uses past observations at the local nodes as well as the fusion nodes; (ii) local nodes transmit only when they detect a change (this reduces interference to the primary and saves energy) (iii) physical layer fusion is exploited in transmitting the data to the fusion node (this saves transmission time in transmitting data from L cognitive radios). Consequently it has been shown to outperform other known algorithms in literature (see [1], [22]). We will provide some comparisons in this paper also.

In the above algorithm we have assumed that the channel from the secondary users to the fusion center has no fading although that can also be taken care of if the channel gains h_l' are known. Also the same DualCUSUM algorithm works if we want to detect the time when the primary stops the transmission (possibly with different parameters).

If the distribution of T is known, then for a single node, Shirayev algorithm is optimal ([12], [25]). One could possibly use that also in our setup at the secondary or fusion nodes. However, especially, in cooperative setup its performance analysis may become intractable (for DualCUSUM and the following modifications itself it is very complicated and we only have an approximate analysis).

B. GLR-CUSUM Algorithm

DualCUSUM assumes that both $f_{1,l}$ and f_0 are available. In Cognitive Radio setup the difficulty is in obtaining perfect knowledge of P_l, the primary's power and the mean noise power

μ_0. In this section we modify the DualCUSUM algorithm to obtain GLR-CUSUM algorithm to take care of the uncertainty in P_l.

We replace the CUSUM algorithm used at the secondary nodes by the GLR algorithm ([12]) which also has certain optimality properties. Let the density of $X_{k,l}$ be f_0 before change and f_θ after change, where θ is a parameter that characterizes density after change. Then the CUSUM algorithm at node l declares change at time $\tau_{\gamma,l}$ defined in (3) (this can be shown to be equivalent to CUSUM explained in steps 1 and 2 of DualCUSUM presented in Section III.A).

$$\tau_{\gamma,l} = \inf\{k: \max_{1\le s\le k}(\sum_{i=s}^{k}\log\frac{f_\theta(X_{i,l})}{f_0(X_{i,l})}) \ > \ \gamma\}. \quad \textbf{(3)}$$

When θ is not known exactly but that $\theta \in \Theta \subseteq \Re$ (where $\Re$ denotes real line), then in GLR we declare change at time

$$\tau_{\gamma,l} = \inf\{k: \max_{1\le s\le k}(\sup_{\theta\in\Theta}\sum_{i=s}^{k}\log\frac{f_\theta(X_{i,l})}{f_0(X_{i,l})}) > \gamma\}. \quad \textbf{(4)}$$

It is assumed that the distribution of fusion receiver noise Z_k is $\sim N(0,1)$. Thus the fusion node can still run CUSUM. Typically the channel gain and the receiver noise within the secondary network would be known to the fusion node ([19]). However if is also not known, then the fusion node can also use GLR.

For the case of mean change, i.e., where pre-change distribution is $N(0,1)$ and the post change distribution is $N(P_l,1)$ the $\sup_\theta(.)$ in (4) is explicitly computed and *max*(.) reduces to

$$W_{k,l} = \max_{1\le s\le k}\{|\sum_{i=s}^{k}X_i|)/(\sqrt{2(k-s+1)\sigma_l^2})\}. \quad \textbf{(5)}$$

If one specifically looks for an increase in mean (which is the valid scenario in cognitive radio as $P_l > 0$), then (5) is replaced as

$$W_{k,l} = \max_{1\le s\le k}\{(\sum_{i=s}^{k}X_i)/(\sqrt{2(k-s+1)\sigma_l^2})\}. \quad \textbf{(6)}$$

If there is minimum post change mean $P_{\min}$, then (6) becomes

$$W_{k,l} = \max_{1\le s\le k}\{(\sqrt{(2\mu_s\sum_{i=s}^{k}X_i - (k-s+1)\mu_{s,k}^2)/2\sigma_l^2})\} \quad (7)$$
$$where\ \mu_{s,k} = \max\{P_{\min}, \sum_{i=s}^{k}X_i/(k-s+1)\}.$$

In this work we have assumed $P_{\min}$ to be zero as this serves as the limiting case for detecting very low SNR which is of interest in Cognitive Radio.

One limitation of the GLR algorithm as stated above is the requirement of an infinite memory. In ([12]) there are techniques suggested to limit the window length as defined below.

We will need the Average Run Length (ARL) which is defined as the mean time to false alarm under the hypothesis that there is no change. Based on [12], for normal approximations, the ARL is related to the threshold γ by

$$\gamma^2 = \log(ARL) + 0.5*\log(\log(ARL)) + \log(K) + o(1)$$
$$\text{where,}\quad K = \pi^{-1/2}\int_0^\infty x\psi^2(x)dx, \quad \textbf{(8)}$$
$$\psi(x) = 2x^{-2}\exp\{-2\sum_{n=1}^{\infty}n^{-1}\Phi(-x\sqrt{n/2})\}$$

and Φ is the distribution of $N(0,1)$. Then the suggested modification is to restrict the computation to a window length of past $M = a\log(ARL)$ decision statistics, where a is a design parameter and chosen such that $a > 2/P_{\min}^2$, where $P_{\min}$ is the lowest change in mean the system needs to detect. Hence in each step, the secondary nodes calculate

$$W_{k,l} = \max_{k-M\le t\le k, t\ge 1}\{(\sum_{i=t}^{k}X_i)/(\sqrt{2(k-t+1)\sigma_l^2})\} \quad \textbf{(9)}$$

The rest of the algorithm at the secondary nodes and at the fusion nodes are same as steps 2-5 of DualCUSUM.

C. MGLR-CUSUM Algorithm

In this section we consider the case, when both noise power and transmit power are unknown to the secondary nodes. Now we assume that the mean noise power μ_0 has a distribution which is not known exactly but is assumed to lie in $[\mu_{ref}/\lambda, \lambda\mu_{ref}]$ where $\lambda > 1$ is an uncertainty parameter. It is easy to see that for the simple threshold based detector (described in the last but one paragraph in Section II and used in [8], [19], [26]), if the primary's transmit power is less than $[(\lambda - 1/\lambda)\mu_{ref}]$ then robust detection is not possible and this is the SNR wall problem.

In this section we modify the GLR-CUSUM algorithm so that it can be made to work under this uncertainty as well. We continue to assume that there is no uncertainty at the fusion node and hence Fusion center can use CUSUM.

In this scenario since both pre and post change parameters are unknown we cannot use DualCUSUM or GLR-CUSUM. However both the pre-change and post-change distributions belong to a single parameter exponential family $f_\theta \sim N(\theta,1)$. To detect a change at the local node, we use log likelihood ratio for the i^{th} hypothesis: the change occurs from θ' to θ'' at time i against the null hypothesis of no change with parameter θ:

$$L_{k,i,l} = \sup_{\theta',\theta''}\{\sum_{s=1}^{i}\log f_{\theta'}(X_{s,l}) + \sum_{s=i+1}^{k}\log f_{\theta''}(X_{s,l})\} - \sup_{\theta}\{\sum_{s=1}^{k}\log f_\theta(X_{s,l})\}. \quad \textbf{(10)}$$

Thus then, at the secondary nodes,

$$\tau_{\gamma,l} = \inf\{k : \max_{1\le i<k} L_{k,i,l} > \gamma\}. \quad \textbf{(11)}$$

We call this algorithm as modified GLR-CUSUM (MGLR-CUSUM). Of course whenever DualCUSUM and GLR-CUSUM can be used those should be preferred because they perform better than MGLR-CUSUM.

The values $\theta', \theta'', \theta$ are chosen to maximize the Likelihood sum. It can be shown for the change in mean case that for a possible change at a point $1 \le i < k$ this reduces to

$$L_{k,i,l} = \{(S_{1,l}^{i})^2 / 2i + (S_{i+1,l}^{k})^2 / 2(k-i) - (S_{1,l}^{k})^2 / 2k\}$$
$$\text{where } S_{i,l}^{n} = \sum_{j=i}^{n} X_{j,l}. \quad \textbf{(12)}$$

Additionally it is shown in [12] that one requires a minimum amount of M^* samples before change. The necessity of this is, as one is looking for a change in mean and if the change occurs very early the algorithm will not be able to detect the change. Also it is easy to see that assuming the change occurs at M^*+1, $\underset{k\to\infty}{Lt}\, E[L_{k,i,l}] \to M^* P^2/2$. Thus for a threshold γ, M^* needs to be much greater than $2\gamma^2 / P_{\min}^2$, where $P_{\min}$ is the minimum power level we want to detect. Further in this algorithm we are either interested in detecting an increase in mean (Primary OFF->ON) or decrease in mean (Primary ON->OFF). Thus the actual implementation in each secondary node is,

$$W_{k,l} = \max_{M^* \le i<k} \sqrt{L_{k,i,l} 1_{\{S_{i+1,l}^{k}/(k-i) > S_{,l1}^{i}/i\}}}\,. \quad \textbf{(13)}$$

The indicator condition in (13) needs to be reversed if we are looking for a decrease in mean. Further it is easy to see that this can be modified to a finite window of $M^* + M$ samples, where M is the window size chosen as in GLR-CUSUM, and $M^* > M$ for robust detection. The complete algorithm is described next.

MGLR-CUSUM Algorithm:

1) At the start of spectrum sensing the fusion center informs all secondary nodes the current assumption about the primary channel, say H_0, i.e., primary is OFF. From now on, the fusion center is only interested in detecting OFF→ON transition.
2) Then each secondary node l, computes the Likelihood ratio $W_{k,l}$ using (13), and is detecting an increase in mean.
3) Secondary node l transmits b at time k, only if $W_{k,l} \ge \gamma$.
4) At the fusion center physical layer fusion is assumed and the steps are same as in DualCUSUM.
5) Once the fusion node declares a change from OFF→ON, it sends this decision instantaneously to the secondary nodes which we assume is received without error. The secondary nodes reset the likelihood ratios, and start the process again, with the condition now being to detect a decrease in mean.

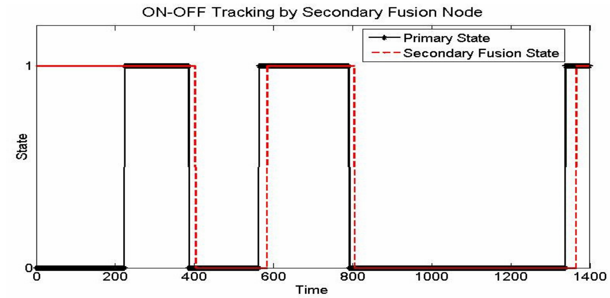

Figure 1: ON-OFF sequence detection by MGLR-CUSUM

Figure 1 shows a sample path of MGLR-CUSUM used by 3 secondary nodes with post change means being {0.7, 0.8, 0.9} to detect the activity of the primary. In the above sequence initially the primary is OFF, and the fusion center assumed it as ON, and hence the first OFF→ON transition is missed by the secondary nodes. But from then on the changes are properly detected. It is noteworthy to add that to have different P_{FA} the threshold γ can be different for detecting OFF→ON from the threshold for detecting ON→OFF.

D. *Performance Comparison*

In this section we compare these algorithms with some other known algorithms. We assume that $\mu_0 = 0$ without loss of generality (as the performance of all the algorithms depends upon the difference in mean). σ_l^2 is 1 at each of the nodes and it is known to each node. Primary's power P_l is different in each of the nodes indicating that the channel gains are different for different l.

To illustrate the benefit of sequential change detection techniques, we compare the performance with some simple slot-by-slot detection schemes ([28]). Here each secondary node compares $\log\{f_{1,l}(X_{k,l}) / f_0(X_{k,l})\} > \lambda_l$ and accordingly decides H_1 or H_0 and transmits 1 or 0 to the fusion center. We assume that the fusion center receives the same instantaneously without error (although in DualCUSUM we are making the more realistic assumption of noise at the fusion center). Fusion center chooses between H_1 or H_0 according to one of the three fusion rules: **OR**: Change is declared if any secondary decides H_1, **AND**: Change is declared if all secondaries decide H_1 and **MAJORITY**: Change is declared if a majority of the secondaries decide H_1.

Parameters used for comparison are as follows. There are 5 nodes with the post change means {0.5, 0.66, 0.7, 0.8, 0.85}. The different parameters for our algorithms are chosen appropriately to meet the desired P_{FA}. The change time T is geometrically distributed with parameter $\rho = 1.25e-3$. For fair comparison with the scenario where the mean noise power is also not known exactly, $(T - M^*)$ is geometric(ρ).

P_{FA}	0.1	0.027	0.01
OR	381.22	1311.7	3140.6
AND	88.93	244.94	491.32
MAJORITY	99.05	294.5	594.3
Dual-CUSUM	9.16	12.41	14.25
GLR-CUSUM	10.79	14.45	16.59
MGLR-CUSUM	11.81	16.09	18.65

Table 1: Performance Comparison of Different algorithms

In Table 1 EDD is expressed in units of slots. As target P_{FA} decreases, gain due to DualCUSUM becomes substantial over the other algorithms. Also one sees that sequential detection algorithms perform much better than other algorithms.

IV. ANALYSIS OF ALGORITHMS

A. *Analysis of DualCUSUM*

The performance analysis of DualCUSUM in this section is based on ([1], [2]). We briefly present the main steps of the analysis and discuss the differences for the present setup from [2]. The analysis for GLR-CUSUM and MGLR-CUSUM will follow the same steps but will differ in some detail. These will be provided in this section.

Let us first consider the events at a secondary node. Let $\tau_{\gamma,l}$ denote the first time the secondary node crosses γ (note that the node dependence is ignored in the notation). Under H_0, i.e., under distribution $f_{0,l}$, it is well known that $\tau_{\gamma,l}$ is exponentially distributed [12]. Let $\lambda_{\gamma,l}$ denote $E[\tau_{\gamma,l}]$. In [2] a renewal equation is presented solving which we can find $\lambda_{\gamma,l}$ numerically.

Once a secondary node crosses the threshold γ it stays above the threshold for a random amount of time. Then it may come back below γ and cross γ a few times before reaching zero. $W_{k,l}$ regenerates at that time and the whole process starts again. The sojourn time of $W_{k,l}$ above γ before touching zero will be called a *Batch*. During this time the cognitive node transmits to the fusion node. This Batch size is analyzed using Brownian motion approximation. Its distribution depends on the overshoot $W_{\tau_{\gamma,l}} - \gamma$ also.

At the fusion node a False Alarm will most likely occur during a batch at some secondary node (when a secondary is falsely transmitting). F_k at the fusion center can also cross β due to noise Z_k itself. It is shown that this time to cross β (when no secondary is transmitting) is also exponentially distributed with parameter λ_0 which can be obtained by the same renewal equation based method as for λ_γ.

The EDD in [2] is obtained as follows. It is shown that post change under $f_{1,l}$, $\tau_{\gamma,l} \sim N(2\sigma_l^2\gamma / P_l^2, 8\sigma_l^4\gamma / P_l^4)$ where P_l is the post change mean at each node. Then a recursive algorithm is used to estimate the times at which the drift changes at the fusion node. Finally law of large numbers and central limit theorem are used to approximate EDD.

The analysis in [1], [2] was for sensor networks where it was assumed that each node has the same distribution. However now it is more reasonable to assume that the channel gains P_l from the primary to each cognitive radio are different. This requires some changes in computing the P_{FA} and EDD compared to the analysis in [2] which we now explain.

The Batchsize distribution at the fusion center now depends on the nodes which are transmitting in the current batch. Thus probability $\widetilde{p}$ of a False Alarm during a batch is

$$\widetilde{p} = \sum_{i=1}^{\infty}\sum_{j=1}^{L} P(l=j)P(\eta=i \mid l=j)P(FA \mid \eta=i)$$

where $P(l=j)$ is the probability that the batch is occurring due to the j^{th} node, η is the batch size and $P(FA \mid \eta = i)$ is the probability of false alarm if the batch size is i. Since $\tau_{\gamma,l}$ are ~exponential($\lambda_{\gamma,l}$), $P[l=j] = \lambda_{\gamma,j} / \sum_{i=1}^{L} \lambda_{\gamma,i}$.

For EDD the individual nodes have, $\tau_\gamma \sim N(2\sigma^2\gamma / P_l^2, 8\sigma^4\gamma / P_l^4)$, where the mean and variance of $\tau_{\gamma,l}$ are different for different secondary nodes. From [3] we compute the mean of the I[th] order statistics of these random variables. This gives the mean time it takes for the drift at the fusion node to become positive ($\triangleq \mu_I$). Then EDD is approximated by $E[\tau(I)] + \beta / \mu_I$. We will see that it is a very good approximation.

With the above modifications we will see that simulations match with the analysis rather closely.

B. *Analysis of GLR-CUSUM*

In this section we analyze the performance of the GLR-CUSUM. As in case of DualCUSUM the time τ_γ to declare change at a secondary node l when there is no change $(T = \infty)$ is known to be exponential ([21]) with parameter $\lambda_{\gamma,l} = 2e^{\gamma^2} / K\gamma$ where K is the constant defined in (8).

The main difference from CUSUM is that the batch size distribution cannot be analyzed as in CUSUM because now the process $\{W_{k,l}\}$ does not have the recursive form of CUSUM. To compute the batch size distribution we again use the Brownian motion approximation $\{W_t\}$ but, unlike before, use crossing by square root boundaries, i.e., probabilities of the form $P[\max_{1\le t\le J}(W_t / \sqrt{t+q}) > \gamma]$. In the following we will ignore the node dependence l in our notation for ease of convenience.

Let $\hat{\tau}$ be the index where the maximum occurs in (9) at time τ_γ. Based on the results from ([7]) we know that $\underset{\gamma\to\infty}{Lt}\, P[\tau_\gamma - \hat{\tau} < 1/\log\tau_\gamma] \to 1$. This means that while W_k depends potentially on all the past samples, at each step it is enough if we consider only a few, say J, immediate past samples only. These samples will most likely contribute to the change. From simulations we found that for a finite γ, it is sufficient to choose $J = 25$.

Thus for analysis purposes, we approximate W_{τ_γ} by $\max_{1\le j\le J}(\bar{S}_1^j/\sqrt{j})$ where $\bar{S}_1^j = \sum_{i=1}^{i=j} X_{\tau_\gamma - i+1}$. This holds true as long as $\hat{\tau} > \tau_\gamma - J$. Since $X_k \sim N(0,1)$, $P[\max_{1\le j\le J}(S_1^j/\sqrt{j}) \ge \gamma]$ is well approximated by $P[\max_{1\le t\le J}(W_t/\sqrt{t}) > \gamma]$ ([7]), where $S_1^j = \sum_{i=1}^{j} X_i$. We approximate $P[\max_{1\le j\le J}(\bar{S}_1^j/\sqrt{j}) \ge \gamma]$ also by this only. This will be justified via simulations.

Let η be the batch size for W_k. Now we compute $P[\eta \le 1]$. At time $\tau_\gamma + 1$, $W_{\tau_\gamma+1} = \max\{\max_{1\le j\le J}(\bar{S}_1^j + X_{\tau_\gamma+1})/\sqrt{j+1}, X_{\tau_\gamma+1}\}$. If $X_{\tau_\gamma+1}$ is zero (we will compensate for this later; note that $E[X_{\tau_\gamma+1}] = 0$), then $W_{\tau_\gamma+1}$ becomes $\max_{1\le j\le J}(\bar{S}_1^j)/\sqrt{j+1}$. Thus $P[\eta \le 1]$ is approximated by $P[\max_{1\le j\le J}(\bar{S}_1^j/\sqrt{j+1}) < \gamma \mid \max_{1\le j\le J}(\bar{S}_1^j/\sqrt{j}) \ge \gamma]$. This in turn is well approximated by $P[\max_{1\le t\le J}(W_t/\sqrt{t+1}) < \gamma \mid \max_{1\le t\le J}(W_t/\sqrt{t}) \ge \gamma]$, which can be computed via the algorithm in [15].

To take care of the effect of $X_{\tau_\gamma+1}$ we approximate $P[\eta \le 1]$ by $P[\max_{1\le t\le J}((W_t + Y_1)/\sqrt{t+1}) < \gamma \mid \max_{1\le t\le J}(W_t/\sqrt{t}) \ge \gamma]$ where $Y_1 \sim N(0,1)$. It is easy to see based on the previous steps, that now $P[\eta \le q]$ can be approximated by $P[\max_{1\le t\le J}((W_t + Y_q)/\sqrt{t+q}) < \gamma \mid \max_{1\le t\le J}(W_t/\sqrt{t}) \ge \gamma]$ where $Y_q \sim N(0,q)$. In these steps, we have inherently assumed that $\max_{1\le j\le q}(\tilde{S}_1^j/\sqrt{j}) < \gamma$ where $\tilde{S}_1^j = \sum_{i=1}^{i=j} X_{\tau_\gamma+q-i+1}$. This is a reasonable assumption especially when γ is large, i.e., in other words we assume there is no immediate upcrossing due to the samples after τ_γ.

Figure 2 shows the batch size pdf via this technique against simulation for $\gamma = 3.5$. We see a very close match.

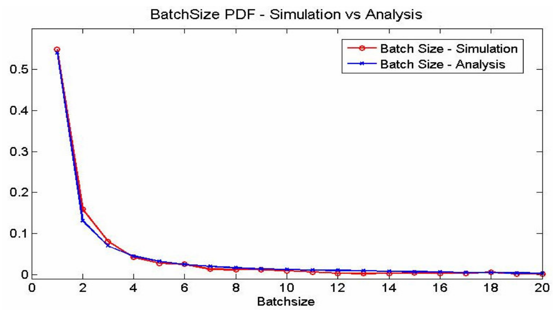

Figure 2: GLR Batchsize PDF simulation vs analysis

The rest of the analysis for P_{FA} is similar to [1], [2]. In GLR-CUSUM unlike in Section III, λ_γ in each node is same because now, P_l has no effect on the statistics of $\{W_k\}$ before change. Hence analysis of [1] carries over easily.

For analyzing EDD of GLR we proceed as follows. It is easy to see that $E[\tau_\gamma] \approx 2\gamma^2/P^2$ ([12]) and we can also approximate $Var[\tau_\gamma]$ by $8\gamma^2/P^4$ as in DualCUSUM (note that $\sigma_l^2 = 1$). The justification is that when γ, P_l are large enough we can show that post change, GLR behaves very similar to CUSUM. Details are omitted due to lack of space. We will see that this provides an accurate enough estimate.

C. Analysis of MGLR-CUSUM

For MGLR it is shown in [12] that τ_γ is exponential with $E[\tau_\gamma] = 2e^{\gamma^2}/(K_{a,a^*}\gamma)$, where K_{a,a^*} is a constant close to K, and depends upon the ratio M^*/M. We obtain K_{a,a^*} via simulations.

For the analysis of batch size for MGLR-CUSUM we can reuse the analysis of GLR-CUSUM, to a large extent. The justification being when M^*, the minimum amount of samples required pre-change, is sufficiently large (which is necessary for cognitive radio to detect small changes in mean), the empirical mean is very close to the actual mean. Thus MGLR-CUSUM behaves very similar to GLR-CUSUM. This is also justified by the results in [7] for change detection in fixed sample scenario which states that the possible change point, under no change, occurs at the very beginning or very end. Since we have a window of M^* samples, change usually occurs at the end. While this entire approach is not very accurate, it provides useful approximation.

For analyzing EDD of MGLR, it can be shown that the lowest value $E[\tau_\gamma] \approx \gamma^2/(P_{\min}^2/2 - \gamma^2/M^*)$ and we continue to use $Var[\tau_\gamma] \approx 8\gamma^2/P^4$. We will see via simulations that this provides a useful approximation for EDD.

D. Simulations

In this section we compare the approximate analysis provided above for the three algorithms with simulation results. We consider two examples $L = 3$ and $L = 5$. When $L = 3$, the post change means are {0.7, 0.8, 0.9} and ρ is chosen as 2.5e-3 and when $L=5$ the post change means are {0.5, 0.66, 0.7, 0.8, 0.85} and ρ is chosen as 1.25e-3. The other parameters for both the examples are $\beta = 11$, $\sigma_l^2 = 1$, $\sigma_M^2 = 1$, $\mu_0 = 0$. The γ and I used in the different cases are provided in Table 2 along with the results.

We observe that the theoretical P_{FA} and EDD for Dual-CUSUM are matching very well. For GLR-CUSUM and MGLR-CUSUM EDD matches very well, although P_{FA} are not so accurate. However considering the complexity of GLR this much accuracy in the cooperative setup seems satisfactory.

	γ	L	I	P_{FA} Simul	P_{FA} Analysis	EDD Simul	EDD Analysis
Dual	5.5	5	5	14e-4	13e-4	18.46	20.22
Dual	6.25	3	3	67e-4	70e-4	18.01	19.24
GLR	2.47	5	5	53e-4	38e-4	18.63	22.88
GLR	2.70	3	3	80e-4	72e-4	19.49	21.95
MGLR	2.55	5	5	43e-4	28e-4	21.08	23.62
MGLR	2.78	3	3	69e-4	51e-4	22.76	24.79

Table 2: Analysis vs Simulation

V. CONCLUSION

We have used the DualCUSUM algorithm presented in [1] for spectrum sensing. DualCUSUM needs the probability densities before and after change. Next we modified it to develop the GLR-CUSUM algorithm to detect change in primary transmission when the received SNR is unknown. The performance of GLR-CUSUM is inferior to DualCUSUM but better than some of the other algorithms. Later on we extend the GLR based algorithm to detect below the SNR wall in a cooperative environment. We have also presented an approximate analysis of these algorithms. Future work includes analyzing these algorithms for other models and optimization for given target constraints.

ACKNOWLEDGEMENT

This research is partially supported by a grant from Boeing Corporation.

REFERENCES

[1] T. Banerjee, V. Kavitha and V. Sharma, “Energy efficient change detection over a MAC using physical layer fusion”, in *Proc of IEEE ICASSP*, Mar 2008, pp 2501-2504.

[2] T.Banerjee and V. Sharma, “Analysis of DualCUSUM: a distributed energy efficient algorithm for change detection”, in *Proc National Communications Conference (NCC)*, Guwahati, Jan 2009. Downloadable from “www.ece.iisc.ernet.in/~vinod”.

[3] H. M. Barakat and Y. H. Abdelkader, “Computing the moments of order statistics from nonidentical random variables”, 2003, *Statistical Methods and applications*, Springer-Verlag.

[4] R.S. Blum, S.A. Kassam and H.V. Poor, “Distributed detection with multiple sensors: Part II-Advanced Topics”, *Proc. IEEE*, Vol. 85, No. 1, pp. 64-79, 1997.

[5] D. Cabric, S. M. Mishra and R. W. Brodersen, “Implementation issues in spectrum sensing for Cognitive Radios”,in *Proc of Asilomar conference on signals, systems and computers*, Nov 2004, Vol 1, pp 772-776.

[6] C. Cordeiro, K. Challapali, D. Birru and N. S. Shankar, “IEEE 802.22: The First Worldwide Wireless Standard based on cognitive radios”, in *Proc IEEE DySPAN*, Nov 2005, pp 328-37.

[7] M. Csorgo and L. Horvath, “Limit Theorems in Change-Point Analysis”, *Wiley Series in Probability and Statistics*, John Wiley and Sons.

[8] G. Ganesan and Y. Li, “Cooperative Spectrum Sensing in cognitive radio, Part I: two user networks”, *IEEE trans. on Wireless Communication*, Vol. 6, June 2007, pp 2204-2213.

[9] S. Geirhofer, L. Tong and B.M.Sadler, “Dynamic Spectrum access in the time domain: Modelling and Exploiting White Space”, *IEEE Commn. Mag*, Vol. 45, pp. 66-72, May 2007.

[10] S. Haykin, “Cognitive Radio: Brain-Empowered Wireless Communications”, *IEEE J. on Selected Areas Commn.*, Vol 23(2), Feb 2005, pp 221-220.

[11] L. Lai, Y. Fan and H.V. Poor, “Quickest Detection in Cognitive Radio: A sequential change detection framework”, *IEEE GlobeCOMM*, Nov 2008.

[12] T. L. Lai, “Sequential changePoint detection in quality control and dynamic systems”, *Journal of the Royal Statistical Society, Series B*, Vol. 57, No. 4 (1995), pp. 613-658.

[13] K. B. Letaief and W.Zhang, “Cooperative Spectrum Sensing” *in Cognitive Wireless Communication Networks*, E. Hossain and V. K. Bhargava, (Eds.), Springer, June 2007.

[14] H. Li, C. Li and H. Dai, “Quickest spectrum sensing in cognitive radio”, in *Proc IEEE CISS, Mar 2008*, pp 203-208.

[15] A. Novikov, V. Frishling and N. Kordzakhia, “Approximations of Boundary Crossing Probabilities for a Brownian Motion”, *Dec. 1999, Journal of Applied Probability*, Vol. 36, No. 4, pp. 1019-1030.

[16] E. S. Page, “Continuous Inspection Schemes”, *Biometrika*, Vol 41, No 1/2 , pp 100-115, June 1954.

[17] A. Parsha, A.A Gohari and A. Sahai, “Exploiting Interference Diversity for Event based Spectrum Sensing”, in *Proc IEEE DySPAN,* Oct 2008.

[18] R. V. Prasad, P. Pawelczak, J. A. Hoffmeyer and H.S. Berger, “Cognitive Functionality in Next Generation Wireless Networks: Standardization Efforts”, *IEEE Communications Magazine*, Vol 46, Apr 2008, pp 72-78.

[19] Z. Quan, S. Qui and A. Sayed, “Optimal Linear Cooperation for Spectrum Sensing in Cognitive Radio Networks”, *IEEE J. on Selected Topics in Signal Processing*, Feb 2008, Vol 2, pp 28-40.

[20] A. Sahai, N. Hoven, S.M. Mishra and R. Tandra, “Fundamental tradeoffs in robust spectrum sensing for opportunistic frequency reuse” in *Proc First Intl Workshop on Tech. and Policy for Accessing Spectrum*, Aug 2006.

[21] D. Siegmund and E. S. Venkatraman, “Using the Generalized Likelihood Ratio Statistic for Sequential Detection of a Change-Point”, *The Annals of Statistics*, 1993, Vol. 23, No. 1, 255-271.

[22] V. Sharma and A. Jayaprakasam, “An Efficient algorithm for cooperative Spectrum Sensing in Cognitive Radio Networks”, in *Proc National Communications Conference (NCC)*, Guwahati, India, Jan 2009. Downloadable from “www.ece.iisc.ernet.in/~vinod”.

[23] R. Tandra and A. Sahai, “SNR Walls for Signal Detection”, *IEEE J. on Selected Topics in Signal Processing*, Vol 2., pp 4-17, Feb 2008.

[24] J. UnniKrishnan and V. V. Veeravalli, “Dynamic Spectrum Access with Learning for Cognitive Radio”, *42nd Asilomar Conf on Signls, Systems and Computers,* Oct-2008, pp. 103-107.

[25] V. V. Veeravalli, “Sequential decision fusion: theory and applications”, *Journal of the Franklin Institute*, 336(2): 301-322, February 1999.

[26] V. V. Veeravalli and J. Unnikrishnan, “Cooperative spectrum sensing for primary detection in Cognitive Radio”, *IEEE Jnl. on Selected Topics in Signal Processing*, Feb 2008, pp. 18-27.

[27] R. Viswanathan and P.K. Varshney, “Distributed Detection with Multiple Sensors I. Fundamentals”, *Proc. IEEE*, Vol. 85, No. 1, pp. 54-63, 1997.

[28] W. Wang, L. Zhang, W. Zou and Z. Zhou, “On the distributed cooperative spectrum sensing for Cognitive Radio”, *Intl. Symposium on Commn. And Info. Technologies*, Oct 2007, pp 1496-1501.

[29] Y. Zhuan, J. Grosspietsch and G. Memik, “Spectrum Sensing using Cyclostationary Spectrum Density for Cognitive Radios”, *IEEE workshop on Signal Processing Systems*, Oct 2007, pp 1-6.