CUSUM Based Distributed Detection in WSNs

Mandar Nadgir†, K. Premkumar†, Anurag Kumar† and Joy Kuri‡

***Abstract*—We consider the problem of quickest intrusion detection occurring at a random time and at a random location using a sensor network. The sensor nodes periodically sample their environment, process the observations (in our case, using a CUSUM-based algorithm) and send a binary–valued *local decision* to the fusion center. The fusion center collects these local decisions and uses a *fusion rule* to process the sensors' local decisions and infer the state of nature, i.e., if an event has occurred or not. Our main contribution is in analyzing two local detection rules in combination with a simple fusion rule. The local detection algorithms are based on the nonparametric CUSUM procedure from sequential statistics. We also propose two ways to operate the local detectors after an alarm. These alternatives when combined in various ways yield several approaches. Our contribution is to provide analytical techniques to calculate false alarm measures, by the use of which the local detector thresholds can be set. Simulation results are provided to evaluate the accuracy of our analysis. As an illustration, we provide a design example. We also use simulations to compare the detection delays incurred in these algorithms.**

Keywords: event detection in wireless sensor networks, distributed detection of a change in distribution, nonparametric CUSUM technique

I. Introduction

Wireless sensor networks (WSNs) comprise a dense ad hoc wireless network of smart sensor nodes (also called *motes*). WSNs can be used to efficiently monitor an area in which they are deployed, and to help in detecting events so that remedial/corrective measures can be taken. The major differences between conventional ad hoc wireless networks and WSNs is the presence of a large number of nodes in WSNs (possibly, randomly deployed), limited power (most applications have sensors that are battery driven with a fixed life time) and limited computation capacity.

Because of the energy constrained nature of the motes in sensor networks, a centralized scheme for *intrusion detection* reduces the lifetime of a network. Decentralized detection schemes which aggregate data, and carry out decision *fusion* at intermediate nodes, enhance the lifetime of the network. We note that the energy needed for computation is significantly less than that needed for communication [1].

Each sensor in the network receives a sequence of observations. Then it sends a sequence of summary messages (or the original data in case of centralized schemes) to the fusion center where a sequential test is carried out to infer the true hypothesis. The main design objective of these sequential tests is to minimize delay with a constraint on false alarm (typically, the probability of false alarm or the mean time time to false alarm).

In this paper, we consider the simplest case of a single event occurring at a random time and a random location in the deployment area $\mathcal{A}$. The event remains in the same place for a sufficiently long time, so that we can confine our attention to the problem of detection and localization of a *stationary* event. Our aim is to develop algorithms both at the node level as well as at the fusion center so that we can detect intrusions as early as possible subject to a constraint on false alarms.

A. Contributions of the Paper:

Based on the nonparametric CUSUM algorithm, we propose two local detector algorithms. A simple fusion rule is proposed. Two variations are proposed for operating the detection process after an alarm. For three combinations of these approaches we provide analytical techniques for calculating false alarm measures, namely, mean time to false alarm,

This work was supported by a project grant from DRDO

†Dept. of Electrical Communication Engg., Indian Institute of Science, Bangalore 560 012; email: mandarmn, anurag, kprem@ece.iisc.ernet.in

‡Centre for Electronics Design and Technology, Indian Institute of Science, Bangalore 560 012; email: kuri@cedt.iisc.ernet.in

or the fraction of time in false alarm. Simulation results are provided to validate the analytical techniques that we develop.

B. *Related Literature:*

There exists a vast body of literature on event detection in sensor networks, starting from the works of Tenny and Sandell [2]. Further work on event detection in decentralized setup was carried out by Tsitsilklis [3]. The sequential approach to the decentralized event detection problem was developed by Veeravalli [4]. Ben-David et al. [5] discuss the use of nonparametric statistics for the event detection in sensor networks. The CUSUM approach was first discussed in statistics by Page [6]. Moustakides [7] discusses the CUSUM algorithm for a decentralized setup (which is used in sensor networks).

C. *Outline of the Paper:*

In Section II, we present the system model for the problem. We propose CUSUM-based local detector algorithms in Section III and the fusion options in Section IV, which are analyzed in Sections V, VI, VII and VIII. To provide a complete picture of the false alarm analysis, we start with the user-defined objectives and provide a design example in Section IX. Section X gives the simulation results for the detection delay incurred for each of the design processes. Finally, we conclude the paper in Section XI.

II. The System Model

We consider a sensor network comprising n nodes (each equipped with a sensor, e.g., acoustic, vibration, or magnetic) uniformly deployed in a region $\mathcal{A}$ for intrusion detection. An intrusion happens at a random time and at a random location. We are interested in detecting the intrusion and identifying its location as early as possible subject to a constraint on some measure of false alarm. Each node makes measurements at a constant sampling rate, uses a local detector to make a local decision, and reports its local decisions to a fusion center. The fusion center makes a global decision on the intrusion.

We consider a discrete-time system and the basic unit of time is one slot. At each slot $k \geqslant 1$, the sensors synchronously sample their environment and obtain the random vector of observations $\mathbf{X}_k = \left(X_k^{(1)}, X_k^{(2)}, \ldots, X_k^{(n)}\right)$. An intrusion (also called an *event*) happens at an unknown time T (called the change time). Before change, the mean signal level at the sensors is zero, and after change the mean signal level at sensor i is $h(\|\ell_i - \ell_e\|)$, where $\ell_i \in \mathcal{A}$ is the location of sensor i, $\ell_e \in \mathcal{A}$ is the location of the event, and $h(.)$ is the received energy level of the event. The observations are then modeled as the output of a transducer, i.e., $X_k^{(i)} = \psi(h(\|\ell_i - \ell_e\|))$. Conditioned on the change time and the change location, the observations are independent across sensors and across time.

At time slot k, sensor i makes a decision $D_k^{(i)} \in \{0,1\}$ based on $\{X_1^{(i)}, X_2^{(i)}, \cdots, X_k^{(i)}\}$ and transmits it to a fusion center. The fusion center receives a vector $\mathbf{D}_k = \left(D_k^{(1)}, D_k^{(2)}, \cdots, D_k^{(n)}\right)$ and maps it into a global decision, $G_k = G_k\left(\mathbf{D}_1, \mathbf{D}_2, \cdots, \mathbf{D}_k\right) \in \{0,1\}$, where 0 means "no change until k" and 1 means "a change has occurred at or before k".

A. *The Motivation For L Coverage*

In a free space, the energy transmitted from a source homogeneously spreads in all directions and hence the received energy has a power law decay. Also, because of foliage and other absorptions in the medium, the received signal energy at a distance d is typically given by

$$h(d) = h_0 e^{-\alpha_2(d-d_0)} \left(\frac{d_0}{d}\right)^{\alpha_1}$$

where $\alpha_1 > 1$, $\alpha_2 > 0$ and h_0 is the reference signal level at a distance d_0 from the source. An event will be sensed by sensors within a distance of d_{min}, only if it is such that $h(d_{min}) = h_{min}$ where h_{min} is the minimum energy level required for sensing. Note that the noise and weaker energy sources will only have an effect up to a distance less than d_{min}.

Given d_{min}, we say that a point in $\mathcal{A}$ is L *covered*, if there are at least L sensors at a distance less than or equal to d_{min} from that point. Let the sensors be deployed in $\mathcal{A}$ such that each point is L covered. Thus, if an event occurs at some point in $\mathcal{A}$, at least L sensors will receive measurements with a positive mean. On the other hand, if there is noise or a weak energy source (i.e., clutter) at some point, then less than L sensors will see the change.

B. A Simple Fusion Rule (SFR)

We propose a simple fusion rule (SFR) at the fusion center based on the notion of L coverage to declare a change at k:

$$G_k = \begin{cases} 1 & \sum_{i=1}^{n} \mathbf{1}_{\{D_k^{(i)}=1\}} \geq L \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

One can obtain a better fusion rule if the sensor locations are known at the fusion center. Also, a sequential fusion rule might perform better than SFR. However, the analysis of such fusion rules is complex and in this paper we confine ourselves to SFR.

III. The Local Decision Algorithms

We consider a local detector employing a non-parametric CUSUM at the sensors.

$$S_k^{(i)} = \left(S_{k-1}^{(i)} + X_k^{(i)} - b \right)^+ \text{ with } S_0^{(i)} := 0 \tag{2}$$

The bias b is such that $\mathbb{E}(X_k^{(i)} - b)$ is negative before change and positive after change. We set $D_k^{(i)} = 1$ if the CUSUM statistic $S_k^{(i)}$ crosses a local threshold $c > 0$. Following the discussion in Section II-A, if an event occurs at $\ell_e \in \mathcal{A}$ then $\mathbb{E}[X_k^{(i)} - b] > 0, \ \forall i$ such that $\|\ell_i - \ell_e\| \leqslant d_{min}$, driving the CUSUM statistic at at least L nodes towards a positive drift, and thus cross the local CUSUM threshold quickly.

In the classical CUSUM procedure, on crossing the threshold, the test is stopped and an alarm is raised. However, in our case, the CUSUM algorithm is allowed to run freely, i.e., it is performed at every node in the region of interest $\mathcal{A}$ even after some nodes have crossed the threshold. This stems out of the need for distributed computation in case of sensor networks, where sensors collaborate in the decision process. Although one sensor may cross the threshold early, one needs to wait until one has “sufficient” information for declaring the change. In addition, a sensor may cross the threshold because of noise, in which case the network should continue making observations, while letting the falsely triggered CUSUM to return to the “untriggered” state.

In the following sections we describe two approaches for using the CUSUM statistic to carry out the local detection.

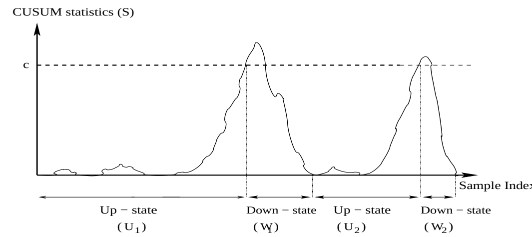

Fig. 1. A sample path for the CUSUM statistic with *Local Detector* 1. Note the way the up-state and the down-state are defined.

A. *Local Detector* 1 *(LD 1)*

LD1 evolves as follows. $S_0 = 0$. Let τ_0 denote the random time at which the CUSUM statistic crosses the threshold c, i.e., $S_{\tau_0} \geq c$ while $S_k < c$ for $0 \leq k < \tau_0$. We then set $S_{\tau_0} = c$ and let the statistic evolve. We say that the node is in the *down-state*, starting from τ_0 until it hits 0. If $\mathbb{E}(X_k - b) > 0$ then with a high probability the statistic will continue to increase and stay above c. On the other hand, if $\mathbb{E}(X_k - b) < 0$ then the statistic will return to 0 with probability 1. We call the up-state intervals as *up-times* and the down-state intervals as *down-times*. In the up-state, the node sends the local decision $D_k = 0$, whereas in the down-state the node sends $D_k = 1$.

Now, in order to analyse the false alarm performance of the system, we consider the situation in which the event never occurs, and, hence, $\mathbb{E}(X_k) = 0$ for all k. It is then clear that the end-points of the down-times (equivalently, the start-points of the up-times) are renewal instants. Also, since we reset S_k to c at the ends of up-times, it is also clear that the alternating sequence of up-times and down-times constitute an alternating renewal process. Let the up-times be denoted by $\{U_k, k \geq 1\}$ and the down-times be denoted by $\{W_k, k \geq 1\}$ (refer to Figure 1).

B. *Local Detector* 2 *(LD2)*

Here, too, the CUSUM algorithm is implemented at each sensor with the statistics S_k being obtained at each sensor using Equation (2). However, in LD2 the S_k is not clipped to c at the beginning of down-state. Here, $D_k = 1$ only if $S_k \geqslant c$. This is shown in Figure 2. Thus the local decisions in LD2 are described as follows. For $k \geq 0$,

$$D_k = \begin{cases} 0 & S_k < c \\ 1 & S_k \geq c \end{cases} \tag{3}$$

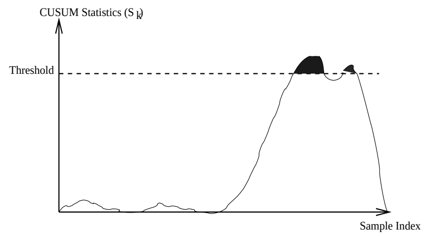

Fig. 2. A sample path for the CUSUM statistics under *Local Detector* 2. The shaded region denotes those samples where the local decision is 1.

IV. Fusion Options and False Alarm Measures

In this section, we study the strategies for operating the system after the fusion center declares a change. We consider the following strategies:

1) *Fusion Option 1 (FO1): Reset the system.* The CUSUM statistic is set to 0 at all the sensors and the count at the fusion center is reset to 0. In this case, the false alarm measure is taken to be *the mean time to false alarm (TFA)*.
2) *Fusion Option 2 (FO2): Continue the process.* The CUSUM statistic is allowed to run without being reset. The false alarm measure is taken to be the fraction of time that the number of sensors with CUSUM statistic greater than 1 is at least L. This is the *fraction of time in false alarm (FROTIFA)*, i.e.,

$$\lim_{m\to\infty} \frac{1}{m} \sum_{k=0}^{m-1} I_{\{(\sum_{i=1}^{n} D_k^i) \geq L\}} \tag{4}$$

where $I\{\cdot\}$ denotes an indicator function.

V. Analysis of Local Detector 1

¿From Section III, it is seen that the local detector can be viewed in terms of an alternating renewal process $\{(U_k, W_k), k \geq 1\}$ (the U_k's are i.i.d., W_k's are i.i.d., and the U_k's are independent of W_k's). If the distributions of U_k and W_k are approximately exponential, then the means $\mathsf{E}(U_k)$, $\mathsf{E}(W_k)$ are sufficient for the analysis of the Fusion Option 1. The approximate exponentiality of U_k has been asserted in [8]. The exponentiality of the Down-time distribution is a heuristic. The fusion process can then be modeled as a continuous time Markov chain (CTMC).

For the LD1 and FO1 combination, we are interested in evaluating the mean time to false alarm (TFA). Hence, we consider the situation, in which the event occurs at infinity, i.e., the sensor observations are i.i.d. across the sensors and over time, with $\mathsf{E}(X_k) - b < 0$, for all k. The analysis of TFA begins with the study of the average run length (ARL) of the CUSUM algorithm at the node level, followed by relating this ARL to the TFA at the fusion center.

TABLE I

Table of mean Up-time values and the Down-time values under the null hypothesis when each sensor uses LD1. Here the bias $b = 0.75$. Here, Sim stands for the simulation results and NE stands for the numerical evaluation.

Thresh	ARL			
c	Up(Sim)	Up(NE)	Down(Sim)	Down(NE)
1.0	18.67	19.03	2.41	2.40
1.25	28.26	28.18	2.72	2.71
1.5	40.78	41.82	3.04	3.03
1.75	62.48	61.92	3.37	3.37
2.0	96.64	91.12	3.72	3.70

A. Analysis of the Up-time

Let $L_\xi(s)$ denote the mean time to cross the threshold when $S_0 = s(< c)$. Note that $L_\xi(s)$ is the solution to the following integral equation:

$$L_\xi(s) = 1 + L_\xi(0)\Phi(b-s) + \int_0^c L_\xi(v)\phi(v+b-s)dv \tag{5}$$

where $\Phi(\cdot)$ is the cdf of X_k, and $\phi(\cdot)$ is the pdf of X_k (under the null hypothesis). Thus $\mathbb{E}U_k$ is obtained by solving for $L_\xi(0)$. Equation (5) is a Fredholm integral equation of the second type. Under the null hypothesis, let $\phi(\cdot)$ be the standard Gaussian for every i and k. With this we can solve the integral equation numerically. We chose the value of $b = 0.75$ and allowed the values of the threshold c to vary from 0.75 to 2.0 in the steps of 0.25. The results are tabulated in Table I.

B. Analysis of Down-time

Let $L_\nu(s)$ be the mean time spent in the down-state if the initial value of the statistic is $s(> 0)$. Using a renewal argument along with Equation (2), and with a few algebraic manipulations, we obtain:

$$L_\nu(s) = 1 + \int_0^\infty L_\nu(v)\phi(v+b-s)\ dv \tag{6}$$

This integral equation is used to evaluate the mean time spent in Down-time, i.e., $L_\nu(0)$ which

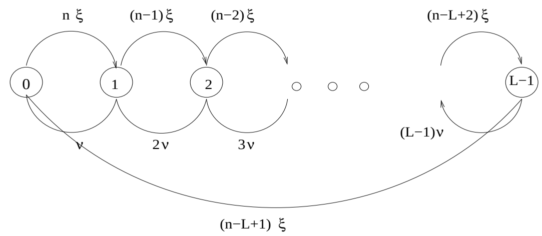

Fig. 3. Markov chain for the fusion rule that count L ones in the area $\mathcal{A}$.

equals $\mathbb{E}W_k$. For $\phi(\cdot)$ being the standard Gaussian density, the results are presented in Table I.

VI. LD1 with Fusion Option 1

In the previous section, we obtained the mean Up and Down times in LD1. With the exponential approximation for the distributions of the Up-times and Down-times, the combination LD1 FO1 can be modeled with a CTMC, the analysis of which yields the the mean time to false alarm (TFA).

For notational convenience, define $\xi = \frac{1}{L_\xi(0)}$, $\nu = \frac{1}{L_\nu(0)}$ and T_{FA} as the mean time to false alarm. Let $N(t)$ denote the state of the fusion statistic at time t; i.e., $N(t)$ is incremented by 1 each time a sensor transitions from up-state to down-state, and is decremented by 1 is the transition is from down-state to up-state. By the exponentiality approximation, we can model $N(t)$ as a CTMC whose state transition diagram is shown in Figure 3. When $N(t) = L-1$, and another sensor's local decision becomes 1, the fusion center declares that a change has been detected and the system is reset.

Since we have the null hypothesis, each transition from $L-1$ to 0 is false alarm. Let $\pi(m), 0 \leq m \leq L-1$, denote the stationary probability distribution of the this CTMC. We observe that the rate of false alarms is $\pi(L-1)(n-(L-1))\xi$. The instants when these false alarms occur are renewal instants. Hence, by the elementary renewal theorem, we conclude that

$$T_{FA} = \frac{1}{\pi_{L-1}(n-(L-1))\xi} \tag{7}$$

The term π_{L-1} is calculated by solving the equation $\pi Q = 0$, where the matrix Q is the transition rate matrix.

Solving for π_{L-1}, we obtain

$$\pi_{L-1} = \frac{1}{\sum_{i=0}^{L-1}(\frac{n-(L-1)}{n-i})\gamma_i} \tag{8}$$

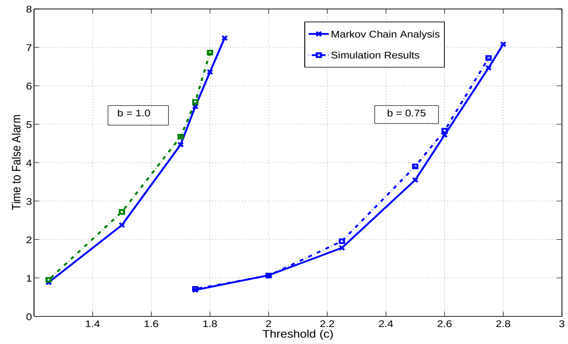

Fig. 4. T_{FA} vs. the threshold c for Local Detector 1 at the sensors and with the Fusion Option 1 at the fusion center, for two bias values, $b = 1.0, 0.75$. Note that the y-axis is $\log_{10}(T_{FA})$.

where, $\gamma_{i+1} = 1 + \gamma_i \frac{(i+1)\nu}{(n-i)\xi}$ and $\gamma_0 = 1$.

Numerical Results: $n = 1000$ sensors were deployed in $\mathcal{A}$. L was chosen to be 40. At each sample instant, each sensor receives a sample of the standard Gaussian distribution. The CUSUM algorithm was implemented for two different values of $b = 0.75, 1$. Numerical results were obtained from Equation 7, and from simulation. The results are plotted in Figure 4.

Observations:

The analytical and simulation results match well, thus justifying the exponentiality approximation made above. Note that the y-axis is in the log scale to the base 10. For $T_{FA} = 10^5$ samples, we can read off the bias and threshold pairs as: $(b = 0.75, c = 2.6)$ and $(b = 1.0, c = 1.75)$. If the sampling interval is 10 seconds, then $T_{FA} = 10^5$ corresponds to a average false alarm rate of 1 every 11 days. For a small change in the value of the threshold, c, there is a large change in the mean time to false alarm. This indicates that the design is quite sensitive to the value of c. In fact, the rate of change is larger for larger values of bias, b. Thus, the user has finer control over T_{FA} with smaller values of the bias. We also see that a small change in the value of the bias (from 0.75 to 1) led to a large change in the value of c (for the same value of TFA). This reflects that the design is sensitive to the bias value.

VII. LD1 with Fusion Option 2

As seen in Section III, with LD1, there are n independent alternating renewal process. The fu-

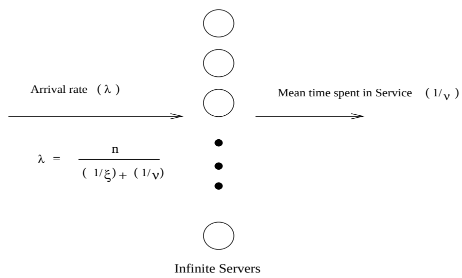

Fig. 5. The $M/G/\infty$ model of the fusion count with LD1 and Fusion Option 2.

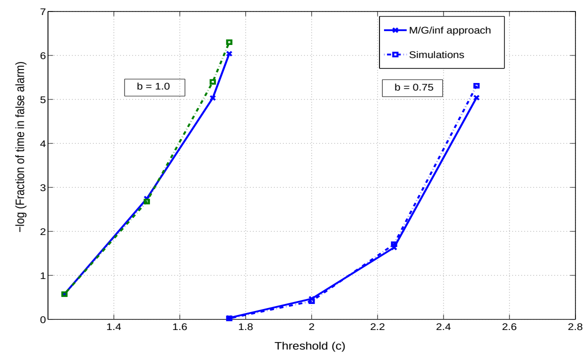

Fig. 6. FROTIFA (on a log scale) vs. the threshold, c, for Local Detector 1 at the sensors with Fusion Option 2, for two bias values, $b = 1.0, 0.75$. Note that the y-axis is $-\log(FROTIFA)$.

sion center maintains a fusion count $N(t)$ which is incremented at and up-to-down transition at a sensor, and decremented at a down-to-up transition. We are interested in the process $N(t)$. The up-to-down transitions at each sensor yields a point process. Each such transition causes the count $N(t)$ to increase. Since $L \ll n$ the aggregate rate of such transitions does not change much even if some sensors are already in the down state. Further, we approximate the superposition of these point processes with a Poisson process. Combining these two approximations, we model the aggregate process of up-to-Down transitions by a Poisson process of rate

$$\lambda_Q \approx \frac{n}{\frac{1}{\xi} + \frac{1}{\nu}} \tag{9}$$

We then see that $N(t)$ is the same as the number of customers in an $M/G/\infty$ queue, with arrival rate λ_Q and holding time distribution the same at that of the Down time, i.e., $\frac{1}{\nu}$. This model can thus be represented as shown in the Figure 5.

We now seek the *fraction of time in false alarm* (FROTIFA). It is well known that the stationary distribution of $N(t)$ is Poisson with mean $\frac{\lambda_Q}{\nu}$. It then follows that FROTIFA is given by

$$\sum_{i=L}^{\infty} e^{-\frac{\lambda_Q}{\nu}} \frac{(\frac{\lambda_Q}{\nu})^i}{i!}$$

Numerical Results: Again we take $n = 1000$, and $L = 40$. Under the null hypothesis the samples are taken to be standard Gaussian. Each sensor runs the LD1 algorithm. The FROTIFA values were evaluated for the bias values $b = 0.75, 1$. The analysis and simulation results are plotted in Figure 6. We observe that the results are very similar to those for LD1 and Fusion Option 1, with T_{FA} as the false alarm measure.

VIII. Analysis of Local Detector 2

In this section we consider the use of LD2 in the sensor, along with Fusion Option 2. Then, as explained earlier, the false alarm measure is FROTIFA. The theory of large deviations helps us relate the FROTIFA measure to the *fraction of time above threshold* (*FOTAT*) measure at each sensor. A target value of FROTIFA yields a target FOTAT. Once a target FOTAT is available, the various design parameters like bias, b, and the threshold, c, can be evaluated.

A. Analysis of LD2 to Obtain FOTAT

In this section, we assume that a bound, ϵ, on the FOTAT at a local detector is available to the user (the next section deals with the procedure to obtain ϵ). The problem then is to relate the bias, b, and threshold, c, at a sensor to the target FOTAT value. We use Chernoff's bound and the affine approximation given by Elwalid et al. [9] to carry out the analysis. Since we are concerned mainly with false alarm analysis, it suffices to analyse the CUSUM process at any one sensor. Hence, as before, we drop the sensor index in the notation.

In LD2 the CUSUM process is not stopped but is allowed to freely evolve based on the measurements. Thus we are interested in analyzing the stochastic process $S_k, k \geq 0$, defined by Equation 2, where the $X_k, k \geq 1$, are i.i.d. with $\mathsf{E}(X_k) - b < 0$. By unraveling the CUSUM recursion we easily

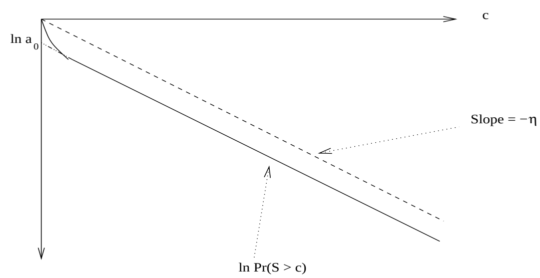

Fig. 7. Typical plot of $\ln Pr(S > c)$ along with a linear bound and an affine approximation.

obtain an alternate expression for the process S_k.

$$S_n = \max_{0 \le k \le n} \left(\sum_{j=(n+1)-k}^{n} X_j - kb \right) \tag{10}$$

1) Using Chernoff's Bound: In this section, we use the Chernoff's bound to obtain an upper bound on the fraction of time during which $S_k \ge c$. Using Equation 10 and the union bound, along with Chernoff's bound, and observing that under the null hypothesis the $X_j, j \ge 1$, are i.i.d., for any $\theta > 0$, we have

$$Pr(S_k \ge c) \le e^{-\theta c} \sum_{k=0}^{n} e^{k(\Gamma(\theta) - \theta b)} \tag{11}$$

where

$$\Gamma(\theta) = \ln \mathsf{E}\left(e^{\theta X_1}\right)$$

If

$$\Gamma(\theta) < \theta b$$

then as the sample index k goes to infinity, it is seen that the summation in Equation 11 converges to a constant, say $\mathcal{K}$. Hence, for large values of k, we can write

$$Pr(S_k \ge c) \le \mathcal{K} e^{-\theta c}$$

In the steady state, writing the marginal of the stationary version of the process as S, we get the following.

$$Pr(S \ge c) \le \mathcal{K} e^{-\theta c}$$

Evidently, to obtain the best bound, we can take $\theta = \eta(b)$ such that $\frac{\Gamma(\eta(b))}{\eta(b)} = b$, thus yielding the following bound on FOTAT at any sensor, under the null hypothesis.

$$Pr(S \ge c) \le \mathcal{K} e^{-\eta(b)c} \tag{12}$$

Now one approximation is to take $\mathcal{K} \approx 1$. Then there exists a linear relationship between $\ln(Pr(S > c))$ and the threshold c. This is plotted as the dashed line in Figure 7.

TABLE II
RESULTS FOR LOCAL DETECTOR 2 SHOWING FOTAT OBTAINED FOR VARIOUS VALUES OF c WITH $b = 0.75$.

Threshold	Simulations	Chernoff	Affine (Elwalid)
c	ϵ_{sim}	ϵ_{Cher}	$\epsilon_{Aff_{Elw}}$
1.8	0.0285	0.0672	0.0270
1.9	0.0246	0.0578	0.0232
2.0	0.0213	0.0498	0.0200
2.1	0.0181	0.0429	0.0172
2.2	0.0158	0.0369	0.0148
2.3	0.0135	0.0317	0.0127
2.4	0.0115	0.0273	0.0109
2.5	0.0100	0.0235	0.0094

As before, we let the samples come from a standard Gaussian distribution. Then, $\Gamma(\theta) = \frac{1}{2}\theta^2\sigma^2$. Using this expression and the definition of $\eta(b)$, we obtain $\eta(b) = \frac{2b}{\sigma^2}$. Setting the bound in Equation 12 equal to ϵ, we obtain $\eta(b) = -\frac{\ln \epsilon}{c}$, yielding the following relation between b, c and the the FOTAT bound, ϵ,

$$bc = -\frac{\sigma^2}{2} \ln \epsilon$$

2) Using an Affine Approximation: As seen from Figure 7 the linear approximation could be quite loose. A more accurate model can be obtained from the affine approximation due to Elwalid et al. [9]. The approximation is

$$Pr(S > c) \approx a_0 e^{-\eta(b)c} \tag{13}$$

where a_0 is approximated as the probability that a single sample exceeds the bias. If the samples come from a standard normal distribution, we have

$$a_0 = Pr(X > b) \approx \frac{\sigma}{b\sqrt{2\pi}} e^{-\frac{b^2}{2\sigma^2}} \tag{14}$$

3) Simulation Setup and Results: We consider a single sensor receiving a sequence of i.i.d. samples that have the standard Gaussian distribution. The simulations were allowed to run for a long time (typically 10^5 samples or more), as compared to the reciprocal of ϵ values. The results are presented in Table II.

4) Observations: It is seen from the table that the results obtained using Chernoff's bound fare poorly and that the affine approximation is quite close to the simulation results. Hence, we make use of only the affine approximation in our work.

B. Analysis of the Fraction of Time in False Alarm (FROTIFA)

We now show how obtain the overall FROTIFA for the Fusion Option 2, with LD2 at each sensor. We make use of results from the theory of large deviations. We begin by using Cramer's theorem, following it with the Bahadur-Rao approximation. We conclude this section with simulation results to compare these two approaches.

1) Using Cramer's Theorem: The sensor level decision processes D_k^i are independent 0-1 for each of which the fraction of time spent in State 1 being ϵ, the FOTAT. Let $D^i, 1 \le i \le n$, denote the steady marginal random variables for the n sensors; these are i.i.d. Bernoulli, with probability of state 1 being ϵ. Since we use the Fusion Option 2, sensor processes are not reset on a global alarm, and the false alarm measure is FROTIFA, which is given by $Pr(\sum_{i=1}^{n} D \ge an)$, where we have defined $a = \frac{L}{n}$. Define the quantities :

$$M(\theta) = E(e^{\theta D^1})$$
$$l(a) = \sup_{\theta}(\theta a - \log M(\theta))$$

Using Cramer's theorem, we have the following standard calculation.

$$\begin{aligned} Pr(\sum_{i=1}^{n} D^i \ge an) &\le \mathbf{E}(e^{\theta \sum_{i=1}^{n}(D^i - a)}) \\ &= e^{-n(\theta a - \log M(\theta))} \\ &\le \inf_{\theta \ge 0} e^{-n(\theta a - \log M(\theta))} \end{aligned}$$

$$Pr(\sum_{i=1}^{n} D^i \ge an) \le e^{-nl(a)} \qquad (15)$$

For Bernoulli random variables, the following expression is well known.

$$l(a) = a \log(\frac{a}{\epsilon}) + (1-a)\log(\frac{1-a}{1-\epsilon})$$

2) Using the Bahadur-Rao approximation : We have seen that conservative results are obtained due to the application of Cramer's theorem. In this section, we investigate a better result, namely, the Bahadur-Rao approximation [10], that can be used instead of the Chernoff's bound. The Bahadur-Rao approximation is given as follows:

$$Pr(\sum_{i=0}^{n} D^i > na) \approx \frac{1}{\theta(a)\sqrt{2\pi n}\sqrt{M''(\theta(a))}} e^{-nl(a)} \qquad (16)$$

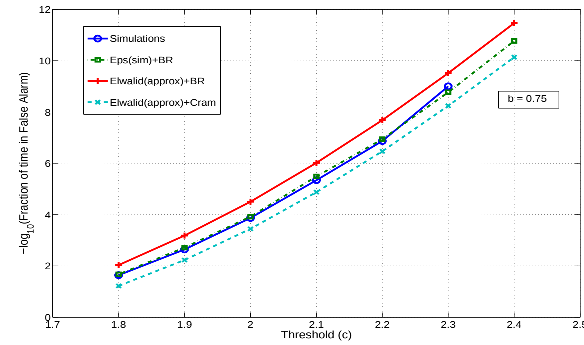

Fig. 8. Results for Local Detector 2 with Fusion Option 2. Fraction of time in false alarm, FROTIFA, (shown a $-\log_{10}$ FROTIFA) for different values of the threshold c; $b = 0.75$. Results from simulations and various approximations are shown.

where $\theta(a)$ achieves the supremum in $\sup_\theta(\theta a - \ln M(\theta))$.

3) Simulation Results: In our simulations, we choose the following parameters: $n = 1000,\ L = 40,\ b = 0.75$. The samples were i.i.d. with standard Gaussian distribution. The simulations were run for a long duration (on the order of 10^9 samples) and compared with the analytical expressions. See Figure 8.

4) Observations:

1) We observe that the FROTIFA is quite sensitive to the values of threshold. Note the log scale on the y-axis.
2) We observe that when the value of ϵ is available (through simulation) and the Bahadur-Rao approximation is used to analyse the fusion rule, the resultant values of FROTIFA match with those obtained entirely through simulations (middle two curves). This shows that the Bahadur-Rao approximation is an excellent for analysing the process at the fusion center.
3) However, if we take the FOTAT (ϵ) from the affine approximations used at the fusion center, the design turns out to be slightly optimistic (the top curve). This can be attributed to the optimistic design at the sensor level (see the last column in Table II, where where the epsilon values are smaller than those actually obtained (first column)).

IX. The User-Defined Objectives and a Design Example

The user defines the objective in terms of the mean time to false alarm, which encapsulates the user's ability to pay a price for false alarms (in the case of an alarm, the user needs to react, which incurs some expense). However, if a very stringent false alarm objective is specified then there could be a large detection delay, which too is detrimental.

A. Converting the User-defined TFA constraint to the FROTIFA constraint

The user defined constraint for false alarms will typically be given in the form of the mean time to false alarm (TFA). However, for analysing the Fusion Option 2 (Section IV)), it is convenient to pose the false alarm constraint in terms of FROTIFA. If, under the null hypothesis, the mean time spent in false alarm is much smaller than the mean time to false alarm (a reasonable assumption in practice), then the following approximation is obtained

$$FROTIFA \approx \frac{1}{TFA}$$

We use this approximation in our design example.

B. A Design Example

The values of n, L and the bias b are related through the event model, the model for propagation of the sensing modality (vibration, acoustics, etc.) and the sensor response model. We assume these as obtained by other means, outside the scope of this work and provide the design of the CUSUM threshold at the sensors. The analysis procedures discussed in Sections VI, VII and VIII are used to obtain the value of the threshold c. The values of $n = 1000$, $L = 40$ and $b = 0.75$ were chosen for this example.

1) Using Local Detector 1 and the Fusion Option 1: The mean time to false alarm is obtained for the algorithm that uses LD1 and Fusion Option 1 using Equation 7. The parameters b, c are implicit in Equation 7 (through π_{L-1} and ξ). However, evaluating the threshold directly from the above equation is not possible as the integral equations involved do not have closed form solutions. Hence, we need to resort to charts (similar to Figure 4). We present the results of the design in Table III.

TABLE III
RESULTS OF THE DESIGN USING LD1 AND FO1, FOR USER DEFINED TFA.

User-Defined TFA	LD1 with FO1	
No. of Samples	Threshold (c)	Design Result (TFA)
10^4	2.54	2×10^4
5×10^4	2.6	6.3×10^4
10^5	2.63	2×10^5
5×10^5	2.67	6.34×10^5
10^6	2.7	1.48×10^6

TABLE IV
RESULTS OF THE DESIGN USING LD1 AND FO2, FOR USER DEFINED TFA.

TFA_{User}	LD1 with FO2		
Samples	$FROTIFA_{target}$	Threshold	Design(TFA)
10^4	10^{-4}	2.42	2×10^4
5×10^4	2×10^{-5}	2.47	1.1×10^5
10^5	10^{-5}	2.5	2.2×10^5
5×10^5	2×10^{-6}	2.54	1.2×10^6
10^6	10^{-6}	2.58	2.51×10^6

C. Using LD1 and FO2

This approach too makes use of the parameters ξ and ν for evaluating the false alarm constraint. Hence, we can evaluate the parameter pair b, c using charts like Figure 6. We present the results of the design in Table IV.

D. Using LD2 with FO2

We make use of the affine approximation at the sensor level and the Bahadur-Rao approximation at the fusion center. The affine approximation was used to obtain the sensor level time above the threshold ϵ.

Using the user-defined T_{FA}, we can thus obtain the parameter ϵ. For our case, the ϵ and the threshold are related as follows:

$$\epsilon = \frac{\sigma}{b\sqrt{2\pi}} e^{-\frac{b^2}{2\sigma^2}} . e^{\frac{-2bc}{\sigma^2}} \qquad (17)$$

Thus, we can directly obtain the value of the threshold. We present the results of the design in Table V.

E. Discussion

¿From Tables III and IV we conclude that the analytical approaches we have developed for LD1+FO1 and for LD1+FO2 result in a slightly conservative design. The TFA actually obtained is slightly larger than the user specification. On the

TABLE V
RESULTS OF THE DESIGN USING LD2 AND FO2, FOR USER DEFINED TFA.

TFA_{User}	LD2 with FO2		
Samples	$FROTIFA_{target}$	Threshold	Design TFA
10^4	10^{-4}	1.97	4×10^3
5×10^4	2×10^{-4}	2.01	1.4×10^4
10^5	10^{-5}	2.04	4×10^4
5×10^5	2×10^{-6}	2.07	1.2×10^5
10^6	10^{-6}	2.1	3×10^5

other hand from Table V we conclude that the affine approximation and Bahadur-Rao approximation based approach for analysing LD2+FO2 yields a design that provides a smaller TFA than the user requirement. In each case it is seen that the TFA is very sensitive to the threshold c, as observed earlier in the paper.

X. DETECTION DELAY: SIMULATION RESULTS

We have only the simulation results for the detection delays. We deploy $n = 1000$ sensors in the area of interest in a grid fashion. The event occurs at a random place (chosen using the uniform distribution), and at random time (chosen geometrically with the parameter $p = 0.0005$). The coverage parameter was chosen to be $L = 40$. For the post-change distributions, the values of μ_i's were calculated as:

$$\mu_i = \frac{b * d_{min}}{d_i}$$

where, $d_{min} = 0.1$ Km and d_i is the distance of the i^{th} sensor from the event. We chose the values of $b = 0.75$ or 1 for our simulations. We evaluate the detection delay in each case for a range of threshold values that were seen to provide a value of the mean time to false alarm on the order of $10^5 - 10^7$ samples.

TABLE VI
DETECTION DELAY. HERE, THR STANDS FOR THRESHOLD, DD STANDS FOR DETECT DELAY, AND LFi STANDS FOR LOCAL DECISION i WITH FUSION RULE i

Bias	Thr(LF1)	DD(LF1)	Thresh (LF2)	DD(LF2)
0.75	2.6	1.13	1.7	1.02
0.75	2.7	1.17	1.75	1.2
0.75	2.75	1.45	1.8	1.39
0.75	2.8	1.54	1.85	1.65
0.75	2.9	1.8	1.9	1.66

A. *Observations*

It is seen that the mean detection delay is in the order of a few samples (0.5 to 2) for the values of threshold that gave the mean time to false alarm of the order of $10^5 - 10^7$ samples.

XI. CONCLUSIONS

We formulated an intrusion detection problem with a user-defined objective of mean time to false alarm. We explored three different approaches based on CUSUM based local detectors at each sensor, and fusion of these local decisions at a fusion center. Our main contribution was to develop analytical techniques for setting the CUSUM threshold for achieving the target TFA. Simulation results were presented to demonstrate how our techniques work in a design example.

Our future work includes the study of an optimal parametric distributed detection scheme within the same framework. Also, we will study ways to design the other parameters, i.e., n, L, and b.

REFERENCES

[1] G. J. Pottie and W. J. Kaiser, "Embedding the internet: Wireless integrated network sensors," *Communications of ACM*, vol. 43, no. 5, pp. 51–58, 2000.

[2] R. Tenney and N. Sandell, "Detction with distributed sensors," *IEEE transactions on Aerospace and Electronics systems*, vol. 17, no. 4, pp. 501–510, 1981.

[3] J. N. Tsitsiklis, "Decentralized detection," *Advances in statistical signal processing Greenwich CT JAI*, pp. 297–344, 1993.

[4] V. V. Veeravalli, "Sequential decision fusion: theory and application," *Journal of the Franklin institute*, pp. 301–322, 1999.

[5] S. Ben-David, T. He, and L. Tong, "Nonparametric approaches to change detectionand estimation in large scale sensor networks." CISS, 2004.

[6] E. S. Page, "Continuous inspection schemes," *Biometrika*, vol. 41, no. 100-115, 1954.

[7] G. V. Moustakides, "Decentralized cusum change detection," 9th International Conference on Information Fusion. Fusion, 2006.

[8] M. Pollack and A. G. Tartakovsky, "On asymptotic exponentiality of the distribution of first exit times for a class of markov processes," *The Annals of Applied Probability*, 2006.

[9] A. Elwalid, D. Heyman, T. V. Lakshman, D. Mitra, and A. Weiss, "Fundamental bounds and approximations for atm multiplexers with applications to video conferencing," *IEEE Journal on Selected Areas in Communications*, vol. 13, no. 6, pp. 1004–1016, 1995.

[10] R. R. Bahadur and R. R. Rao, "On deviations of sample mean," *The Annals of Mathematical Statistics*, vol. 31, no. 4, pp. 1015–1027, 1960.