BYZANTINE FAULT TOLERANT DISTRIBUTED QUICKEST CHANGE DETECTION

ERHAN BAYRAKTAR AND LIFENG LAI

Abstract. We introduce and solve the problem of Byzantine fault tolerant distributed quickest change detection in both continuous and discrete time setups. In this problem, multiple sensors sequentially observe random signals from the environment and send their observations to a control center that will determine whether there is a change in the statistical behavior of the observations. An unknown subset of sensors are compromised and will send arbitrarily modified and even artificially generated signals to the control center. It is shown that the performance of the so-called CUSUM statistic, which is optimal when all sensors are honest, will be significantly degraded in the presence of even a single dishonest sensor. In particular, instead of in a logarithmically the detection delay grows linearly with the average run length (ARL) to false alarm. To mitigate such a performance degradation, we propose a fully distributed low complexity detection scheme. We show that the proposed scheme can recover the log scaling. We also propose a centralized group-wise scheme that can further reduce the detection delay.

1. Introduction

In the quickest change detection problem, one observes a sequence of observations, whose probability density function (pdf) might change at an unknown time [15]. The goal in this problem is to detect the presence of such a change with a minimum delay under certain false alarm constraints. This type of problem has a broad range of potential applications, such as quality control [16], security [21, 18], wireless communications [7], and chemical or biological attack detection, etc.

In recent years, motivated by the rapid development of wireless sensor networks, distributed quickest detection has attracted much interest [25, 26, 11, 14, 24, 22, 20, 23, 3]. In such networks, multiple sensors are deployed to facilitate the detection of abnormal activities. These sensors will send observations to a fusion center which will then make final detect decisions [27, 2]. In the existing distributed quickest change detection setup, the sensors are assumed to always report true observations. In certain applications, especially in the attack detection applications, sensors might be compromised. As a result, the compromised sensors might send modified or even artificially generated observations. In this paper, we consider the quickest detection problem under such Byzantine attacks. The problems studied in this paper are related to the Byzantine fault tolerance problems [8]
in which a component will fail in an arbitrary manner (instead of just stopping functioning). Our goal is to design Byzantine fault tolerant quickest detection schemes.

More specifically, we consider a setup with \( N \) sensors, an unknown subset of which might be compromised by adversaries. These sensors observe signals sequentially from the environment. If a sensor is honest, it will send its observations to a control center. If a sensor is compromised, it will send modified observations to the control center. We do not make any assumption on how the attacker will modify its observations. We consider a non-Bayesian setup in which the statistical behavior of the random observations from the environment will change at an unknown but fixed time. The goal of the control center is detect the presence of such a change based on the signals received from all sensors.

If all sensors are honest, it is well known that the cumulative sum (CUSUM) strategy is optimal under Lorden’s setup [9]. We first show that the performance of CUSUM will be significantly degraded even if only one sensor is compromised. In particular, using a simple attack, the attacker can make the detection delay of CUSUM scale linearly with the average run length (ARL) to false alarm (see Proposition 3.1), while the detection delay of CUSUM scales in a logarithmically with ARL to false alarm when there is no attacker.

To overcome such a performance degradation, we propose a low complexity, low communication overhead detection scheme. In this scheme, we ask each sensor to run a CUSUM locally using its own observed signal, and send one bit of indication to the control center once its CUSUM stops. The control center will raise a final alarm once it receives alarms from at least two sensors. We call this scheme the “second-alarm” strategy/scheme. Using tools from order statistics [4], we provide an upper-bound on the detection delay and an lower-bound on the ARL to false alarm that are valid under arbitrary attack strategies. We show that the derived upper-bound on the detection delay scales logarithmically with that of the lower-bound on the ARL to false alarm; see Theorems 3.2 and 4.1. This implies that the proposed second-alarm strategy successfully recovers the logarithmic scaling. Furthermore, this scheme can be implemented in a fully distributed manner.

To further reduce the detection delay, we propose a group-wise strategy. In this group-wise strategy, we divide sensors into three groups. Each group will run a CUSUM using signals received in its group and raise an alarm once its CUSUM stops. The control center then raises the final alarm once at least two groups raise alarms. We show that the detection delay of the group-wise strategy scales in a logarithmically with the ARL to false alarm. Furthermore, the pre-log factor of the group-wise strategy is smaller; see Theorems 3.6 and 4.2. The main intuition for the deduction of the detection delay in the group-wise strategy is that CUSUM in each group can raise an alarm faster than CUSUM in each individual sensor. The disadvantage is that the group-wise strategy is not amenable to distributed implementation.

The remainder of the paper is organized as follows. In Section 2, we introduce the models discussed in the paper. In Section 3, we study the continuous-time Brownian motion model. In Section 4, we discuss the discrete-time model.
2. Problem Formulation

Consider a system with $N$ sensors that are deployed to monitor the environment. We assume that one sensor might be compromised. The schemes discussed in the paper can be properly modified to handle multiple compromised sensors. We use $n^c$ to denote the index of the compromised sensor. The value of $n^c$ is unknown. The behavior of the compromised and honest sensors will be specified in the sequel. We assume that the sensors are connected with a fusion center, which will make detection decisions based on the observations from the sensors. We consider two different models: 1) a continuous-time Brownian motion model and 2) a discrete-time model.

![Figure 1. Model](image)

**Figure 1. Model**

2.1. The Continuous-time Brownian Motion Model. In this model, sensor $n \in \{1, \ldots, N\}$ observes $\{\xi_t^{(n)}; t \geq 0\}$:

$$\xi_t^{(n)} = \mu(t - \tau)^+ + W_t^{(n)},$$

where $\tau$ is an unknown constant at which signals observed at all sensors change, $\mu$ is the signal strength at each node after the change, $\{W_t^{(n)}\}_{n \in \{1, \ldots, N\}}$ are independent standard Brownian motions. Hence, in our model, we consider a non-Bayesian setup with the change occurring simultaneously across all sensors.

Let $\tilde{\xi}_t^{(n)}$ be the signal sensor $n$ reports to the fusion center. If sensor $n$ is honest, $\tilde{\xi}_t^{(n)} = \xi_t^{(n)}$. For the compromised sensor $n^c$, $\tilde{\xi}_t^{(n^c)}$ depends on the attacker’s attack strategy $\varsigma$, which can use the information $\xi_t^{(n^c)}$ observed by the attacker. We do not make any assumption on the attack strategy. Figure 1 illustrates the model under consideration.

Let $\mathcal{F} = \bigcup_{t \geq 0} \mathcal{F}_t$, where $\mathcal{F}_t = \sigma \left\{ \left( \tilde{\xi}_s^{(1)}, \ldots, \tilde{\xi}_s^{(N)} \right); s \leq t \right\}$. The goal of the fusion center is to detect the presence of a change in the environment based on the processes $(\tilde{\xi}_t^{(1)}, \ldots, \tilde{\xi}_t^{(N)})$. Let $T$ be a stopping rule that the fusion center uses to raise an alarm.

With a little of abuse of notation, we use the following modified Lorden’s performance index as the delay measure

$$d(T) = \sup_{\tau, n^c, \varsigma} \mathbb{E}_{\tau, n^c, \varsigma}[\mathbb{E}_{\mathcal{F}_\tau}[(T - \tau)^+]].$$
Here \( \text{esssup} \) denotes the \( L^\infty \) norm, \( E_{\tau,n^c,s}(T-\tau^+|F_\tau) \) is the expectation under the measure when the change occurs at time \( \tau \), the compromised sensor is \( n^c \) and the compromised sensor uses the attack strategy \( s \). We note that for some attack strategies, for example when the attack strategy varies arbitrarily over time, the expectation \( E_{\tau,n^c,s}(T-\tau^+|F_\tau) \) might be ill-defined.

We also have the following average run length (ARL) to false alarm measure

\[
\text{ARL}(T) = \inf_{n^c,s} E_{\infty,n^c,s}[T].
\]

Here \( E_{\infty,n^c,s}[T] \) denotes the expectation under the measure when the change does not occur, the compromised sensor is \( n^c \) and the compromised sensor uses the attack strategy \( s \). Similar to the discussion above, this expectation might be ill-defined.

With the above defined metrics, we aim to solve the following optimization problem

\[
\inf_T d(T) \quad \text{such that } \text{ARL}(T) \geq \gamma.
\]

(2.1)

In other words, we aim to design a stopping rule that minimizes the detection delay while making sure that ARL to false alarm is larger than a given threshold \( \gamma \).

2.2. Discrete-time Model. In the discrete-time model, we observe \( N \) mutually independent sequences \( \{\xi_k^{(n)}; k \geq 1\}, n \in \{1, \cdots, N\} \), in which \( n \) is the sensor index and \( k \) is the time index, with the following probability density functions (pdf)

\[
\xi_k^{(n)} \sim \begin{cases} 
    f_0, & k \leq \tau, \\
    f_1, & k > \tau.
\end{cases}
\]

Here \( f_0 \) is the pdf of each observation before the change and \( f_1 \) is the pdf of each observation after the change.

The remainder of the setup is similar to the setup in Section 2.1. In particular, let \( \tilde{\xi}_k^{(n)} \) be the signal sensor \( n \) reports to the fusion center. If sensor \( n \) is honest, \( \tilde{\xi}_k^{(n)} = \xi_k^{(n)} \). For the compromised sensor \( n^c \), \( \tilde{\xi}_k^{(n^c)} \) depends on the attacker’s attack strategy \( s \), which can use the information \( \xi_k^{(n^c)} \) observed by the attacker. Again, we do not make any assumption on the attack strategy.

Let \( F = \bigcup_{k \geq 1} F_k \), where \( F_k = \sigma(\{\tilde{\xi}_i^{(1)}, \cdots, \tilde{\xi}_i^{(N)}\}; i \leq k) \). The goal of the fusion center is to detect the presence of a change in the environment based on the processes \( \{\tilde{\xi}_k^{(1)}, \cdots, \tilde{\xi}_k^{(N)}\} \). Let \( T \) be a stopping rule that the fusion center uses to raise an alarm.

Similar to the Brownian motion case, we use the following modified Lorden’s performance index as the delay measure

\[
d(T) = \sup_{\tau,n^c,s} \text{esssup} E_{\tau,n^c,s}(T-\tau^+|F_\tau).
\]

We also have the following average run length (ARL) to false alarm measure

\[
\text{ARL}(T) = \inf_{n^c,s} E_{\infty,n^c,s}[T].
\]
Again, we aim to solve the following optimization problem

\[
\inf_T d(T)
\]

such that \(\text{ARL}(T) \geq \gamma\).

3. Continuous-time Brownian Motion Model

In this section, we will first discuss the impact on the CUSUM test of the presence of a single attacker. We show that, instead of scaling logarithmically, the detection delay of CUSUM scales linearly with ARL to false alarm under a simple attack. We then propose a fully distributed scheme that allows us to recover the logarithmic scaling. We further design a centralized scheme that allows us to further reduce the detection delay.

3.1. Impact on the CUSUM test. We first show that the presence of even a single attacker has significant impact on the performance of CUSUM.

When all nodes are honest, i.e., \(\tilde{\xi}^{(n)}_t = \xi^{(n)}_t\), \(\forall n\), the non-Bayesian quickest change detection problem under Lorden’s performance index has been well understood. The optimal solution is the continuous-time version of Page’s CUSUM stopping rule; see e.g. [1, 17]. In particular, if we define

\[
y_t = u_t - m_t,
\]

with

\[
u_t = \sum_{n=1}^{N} \mu^{(n)}_t - \frac{1}{2}N\mu^2 t,
\]

\[
m_t = \inf_{0 \leq s \leq t} u_s,
\]

then, the CUSUM stopping rule is

\[
T_\nu = \inf \{t \geq 0; y_t \geq \nu\},
\]

where \(\nu\) is a threshold such that

\[
\text{ARL}(T_\nu) = E_\infty[T_\nu] = \frac{2(e^{-\nu} - \nu - 1)}{N\mu^2} = \gamma.
\]

is optimal for (2.1) for the case when there is no adversary. For this scheme, the detection delay is

\[
d(T_\nu) = \frac{2(e^{-\nu} + \nu - 1)}{N\mu^2}.
\]

From here, we know that when the ARL constraint \(\gamma \uparrow \infty\), we have

\[
d(T_\nu) = \frac{2}{N\mu^2} \left[ \log(\text{ARL}(T_\nu)) + \log \frac{N\mu^2}{2} - 1 + o(1) \right].
\]

This means that the detection delay scales logarithmically with ARL.

Now suppose one of the sensors is compromised, but the fusion center still employs the CUSUM test in (3.2) to detect the change, then a simple attack will significantly degrade the performance
of CUSUM. In this attack, the attacker chooses to set $n^c = 1$, and generates $\bar{\xi}_t^{(1)}$ according to the following dynamics irrespective of $\xi_t^{(1)}$
\begin{equation}
\bar{\xi}_t^{(1)} = Nt + W_t^{(1)}, \quad t \geq 0.
\end{equation}

We use $s^*$ to denote this particular attack strategy, and use $P_{0,1,s^*}$ to denote the probability measure when the change point occurs at time 0, and $P_{\infty,1,s^*}$ to denote the probability measure when the change point never occurs.

The impact of this simple attack on the performance of the CUSUM test (3.2) is computed in the following proposition.

**Proposition 3.1.** Under the attack strategy specified in (3.4), the ARL to false alarm and the detection delay of CUSUM (3.2) are:
\begin{equation}
ARL(T_\nu) \leq E_{\infty,1,s^*}[T_\nu] = \frac{(e^{-\nu} + \nu - 1)}{N\mu^2/2},
\end{equation}
and
\begin{equation}
d(T_\nu) \geq E_{0,1,s^*}[T_\nu] \geq \frac{(e^{-\nu} + \nu - 1)}{(3N/2 - 1)\mu^2},
\end{equation}
respectively. As a result, instead of logarithmically, the detection delay of CUSUM scales linearly with ARL to false alarm under a simple attack.

**Proof.** First let us introduce
\begin{equation*}
u_t = \sum_{n=1}^{N} \mu \tilde{\xi}_t^{(n)} - \frac{1}{2} N \mu^2 t = \mu \sum_{n=1}^{N} \tilde{\xi}_t^{(n)} - \frac{1}{2} N \mu^2 t.
\end{equation*}

Under probability measure $P_{0,1,s^*}$, the process $\xi_t = \sum_{n=1}^{N} \tilde{\xi}_t^{(n)}$ can be written as
\begin{equation*}
\xi_t = (2N - 1)\mu t + \sqrt{N}W_t,
\end{equation*}
for some standard Brownian motion $W_t$. As a result, under $P_{0,1,s^*}$
\begin{equation*}
u_t = (3N/2 - 1)\mu^2 t + \sqrt{N}\mu W_t.
\end{equation*}

Now following the techniques introduced in [13], we will compute $E_{0,1,s^*}[T_\nu]$. Let us denote $f(y) = y + e^{-y} - 1$.

It is easy to check that
\begin{equation*}
f'(y) = 1 - e^{-y}, \quad f''(y) = e^{-y}, \quad f'(y) + f''(y) = 1, \quad f'(0) = f(0) = 0,
\end{equation*}
and $f''(y) > 0, \forall y$.

Applying Itô’s rule, we obtain
\begin{align*}
f(y_t) - f(0) &= \int_{0}^{t} f'(y_t)(du_t - dm_t) + \int_{0}^{t} \frac{N}{2} \mu^2 f''(y_t)dt \\
&= \int_{0}^{t} (3N/2 - 1)\mu^2 f'(y_t)dt + \int_{0}^{t} \sqrt{N}\mu f'(y_t)dW_t - \int_{0}^{t} f'(y_t)dm_t + \int_{0}^{t} \frac{N}{2} \mu^2 f''(y_t)dt.
\end{align*}
Using \( f'(0) = 0 \) we obtain that \(-\int_0^t f'(y_t) dm_t = 0\). Hence, we are left with

\[
 f(y_t) - f(0) = \int_0^t (3N/2 - 1)\mu^2 f'(y_t)dt + \int_0^t \sqrt{N}\mu f'(y_t)dW_t + \int_0^t \frac{N}{2}\mu^2 f''(y_t)dt
\]

\[
 \leq \int_0^t (3N/2 - 1)\mu^2 f'(y_t)dt + \int_0^t \sqrt{N}\mu f'(y_t)dW_t
\]

\[
 \overset{(b)}{=} (3N/2 - 1)\mu^2 t + \int_0^t \sqrt{N}\mu f'(y_t)dW_t,
\]

in which \((a)\) in the above equation is due to the fact that \( f''(y) > 0 \), and \((b)\) is due to the fact that \( f'(y) + f''(y) = 1 \). Now, \((3.5)\) easily follows. The computation of \( E_{\infty,1,\nu^*}[T_{\nu}] \) is similar to that of \([13]\). \(\square\)

3.2. A Fully Distributed Byzantine Attack Resistant Change Detection Scheme. In this section, we propose an intuitive scheme that enables us to recover the logarithmic scaling. In this scheme, the fusion center first runs \( N \) independent CUSUMs, \( T_h^{(n)} \), \( n = 1, \cdots, N \), one for each signal \( \xi_t^{(n)} \) received from sensor \( n \):

\[
 T_h^{(n)} = \inf\{t \geq 0; y_t^{(n)} \geq h\},
\]

in which

\[
 y_t^{(n)} = \mu \xi_t^{(n)} - \frac{1}{2}\mu^2 t - \inf_{s \leq t} \left( \mu \xi_s^{(n)} - \frac{1}{2}\mu^2 s \right).
\]

We call \( T_h^{(n)} \) the time at which sensor \( n \) raises an alarm. In our scheme, the fusion center will ignore the first alarm from sensors, and will raise the final alarm once a second sensor raises an alarm. Let \( T_h \) be the time when the fusion center raises the final alarm. From the description above, we know that \( T_h \) is the same as the second order statistics of \( N \) i.i.d. random variables \( T_h^{(1)}, \cdots, T_h^{(N)} \).

Following the notation in order statistics \([4]\), we use \( T_{(2),N} \) to denote the second order statistics of \( N \) random variables \( T_h^{(1)}, \cdots, T_h^{(N)} \). As a result, the time at which the fusion center raises an alarm is \( T_h \overset{\Delta}{=} T_{(2),N} \).

Now, we analyze the detection delay \( d(T_h) \) and ARL to false alarm \( ARL(T_h) \). We have the following result:

**Theorem 3.2.** Under any attack strategy, the detection delay of \( T_h \) scales with ARL at most logarithmically:

\[
 d(T_h) \leq \frac{4}{\mu^2} \left( \log(ARL(T_h)) + \log \left( \frac{(N - 1)\mu^2}{2} \right) - 1 + o(1) \right).
\]

**Proof.** We first provide a lower bound on ARL to false alarm that is valid for any attacker’s strategy. From the attacker’s perspective, to reduce ARL to false alarm of the proposed scheme, it should make the compromised sensor to raise an alarm as soon as possible. The proposed scheme will then raise a false alarm when one of the remaining \( N - 1 \) honest sensors raises an alarm under the probability measure \( P_\infty \). Since all sensors are identical, for the purpose of performance evaluation, we can simply assume that sensor \( N \) is compromised. Let \( T_{(1),N-1} \) be the first order statistics of
To proceed, let us recall the following result from [10]:

\[ T^{(1)}_h, \cdots, T^{(N-1)}_h, \text{namely } T^{(1),N-1}_h = \min \left\{ T^{(1)}_h, \cdots, T^{(N-1)}_h \right\}. \]

From the discussion above, we have

\[ \text{ARL}(T_h) \geq E_\infty \{ T^{(1),N-1}_h \} . \]

Since \( T^{(1),N-1}_h \) is a positive random variable, we have

\[
\begin{align*}
\text{ARL}(T_h) &\geq E_\infty \{ T^{(1),N-1}_h \} \\
&= \int_0^\infty P_\infty (T^{(1),N-1}_h \geq t) \, dt \\
&= \int_0^\infty P_\infty \left( \min \left\{ T^{(1)}_h, \cdots, T^{(N-1)}_h \right\} \geq t \right) \, dt \\
&= \int_0^\infty \left[ P_\infty (T^{(1)}_h \geq t) \right]^{N-1} \, dt.
\end{align*}
\]

The last equality is due to the fact that for those sensors not affected by the attacker, \( T^{(n)}_h \) are i.i.d. random variables.

Next, we provide an upper-bound on the detection delay that holds for any strategy of the attacker. Clearly, to increase the detection delay of the proposed scheme, the attacker should use a strategy that does not raise an alarm. In this case, the fusion center will raise an alarm only when at least two honest sensors raise alarms under the probability measure \( P_0 \). Again, for the performance evaluation purpose, we can simply assume that sensor \( N \) is compromised.

Let \( T^{(2),N-1}_h \) be the second order statistics of \( T^{(1)}_h, \cdots, T^{(N-1)}_h \). Based on the discussion above, we have

\[
d(T_h) \leq E_0 \{ T^{(2),N-1}_h \}
\]

\[
= \int_0^\infty P_0 (T^{(2),N-1}_h \geq t) \, dt
\]

\[
= \int_0^\infty P_0 \left\{ \text{at most 1 of } T^{(1)}_h, \cdots, T^{(N-1)}_h \text{ is less than } t \right\} \, dt
\]

\[
= \int_0^\infty P_0 \left\{ \text{none of } T^{(1)}_h, \cdots, T^{(N-1)}_h \text{ is less than } t \right\} \, dt
\]

\[
+ \int_0^\infty P_0 \left\{ 1 \text{ of } T^{(1)}_h, \cdots, T^{(N-1)}_h \text{ is less than } t \right\} \, dt
\]

\[
= \int_0^\infty P_0 \left( \min \left\{ T^{(1)}_h, \cdots, T^{(N-1)}_h \right\} \geq t \right) \, dt + \int_0^\infty P_0 \left\{ 1 \text{ of } T^{(1)}_h, \cdots, T^{(N-1)}_h \text{ is less than } t \right\} \, dt
\]

\[
= \int_0^\infty \left[ P_0 (T^{(1)}_h \geq t) \right]^{N-1} \, dt + \int_0^\infty (N-1)(1-P_0 (T^{(1)}_h \geq t)) \left[ P_0 (T^{(1)}_h \geq t) \right]^{N-2} \, dt
\]

To proceed, let us recall the following result from [10]:

\[
P_0 (T^{(n)}_h \geq t) = 2e^{-\frac{h}{2}} \sum_{k \geq 1} u(\phi_k^{(n)}) e^{-\frac{\nu^2 t}{8 \cos^2 (\phi_k^{(n)})}}, \tag{3.6}
\]

\[
P_\infty (T^{(n)}_h \geq t) = 2e^{-\frac{h}{2}} \sum_{k \geq 1} u(\theta_k^{(n)}) e^{-\frac{\nu^2 t}{8 \cos^2 (\phi_k^{(n)})}} + 2e^{-\frac{h}{2}} v(\eta^{(n)}) e^{-\frac{\nu^2 t}{8 \cosh^2 (\eta^{(n)})}}, \tag{3.7}
\]
It now follows from (3.8) and (3.9) that
d
and the constants \( \phi_k^{(n)} \), \( \theta_k^{(n)} \), and \( \eta^{(n)} \) are defined as the solutions of
\[
\tan \phi_k^{(n)} = -\frac{2\phi_k^{(n)}}{h} < 0, \quad \tan \theta_k^{(n)} = \frac{2\theta_k^{(n)}}{h} > 0, \quad \text{and} \quad \tanh \eta^{(n)} = \frac{2\eta^{(n)}}{h} > 0,
\]
respectively. With these results, we continue our estimations.
\[
\begin{align*}
\text{ARL}(T_h) & \geq \int_0^\infty \left[P_\infty(T_h^{(1)} \geq t)\right]^{N-1} dt \\
&= \int_0^\infty \left[2e^{-\frac{h}{N}} \sum_{k=1}^{N-1} u(\theta_k^{(n)}) e^{-\frac{\mu^2 t}{8 \cos^2(\phi_k^{(n)})}} + 2e^{-\frac{h}{N}} v(\eta^{(n)}) e^{-\frac{\mu^2 t}{8 \cosh^2(\eta^{(n)})}}\right]^{N-1} dt \\
& \quad \overset{(a)}{=} \frac{2}{(N-1)\mu^2} [e^h + \text{“lower exponents”}],
\end{align*}
\]
in which (a) in (3.8) is true due to (80) of [5].

As for \( E_0\{T_{(2),N-1}\} \), it is obvious that \( P_0(T_{(2),N-1} \geq t) \) is a non-increasing function of \( N \). Hence from (3.2), we have
\[
\begin{align*}
d(T_h) & \leq E_0\{T_{(2),N-1}\} \\
& = \int_0^\infty \left[P_0(T_h^{(1)} \geq t)\right]^{N-1} dt \\
& \quad + \int_0^\infty (N-1)(1-P_0(T_h^{(1)} \geq t)) \left[P_0(T_h^{(1)} \geq t)\right]^{N-2} dt \\
& \leq \int_0^\infty [P_0(T_h^{(1)} \geq t)]^2 dt + \int_0^\infty 2(1-P_0(T_h^{(1)} \geq t))P_0(T_h^{(1)} \geq t) dt \\
& = 2 \int_0^\infty P_0(T_h^{(1)} \geq t) dt - \int_0^\infty [P_0(T_h^{(1)} \geq t)]^2 dt \\
& = \frac{4(e^{-h} + h - 1)}{\mu^2} - \int_0^\infty [P_0(T_h^{(1)} \geq t)]^2 dt \\
& \leq \frac{4(e^{-h} + h - 1)}{\mu^2}.
\end{align*}
\]
It now follows from (3.8) and (3.9) that \( d(T_h) \) grows logarithmically with ARL\( (T_h) \).

\[ \square \]

**Remark 3.3.** The bounds derived in the proof are valid for any strategy of the attacker, who can use an arbitrarily complicated attack strategy and can arbitrarily change its behavior over time.

**Remark 3.4.** This scheme is amenable to a fully distributed implementation with very little communication overhead. In particular, each sensor can run \( T_h^{(n)} \) locally, and send an alarm (one bit) to the fusion center once \( y_t^{(n)} \) is larger than the threshold \( h \). The fusion center will raise a final alarm after receiving the second alarm from the sensors.

**Remark 3.5.** The proposed scheme also works for the scenario in which the Brownian motion observed by each sensor has different drift and volatility coefficients. The analysis follows the same steps as above but with more complicated computations.
3.3. A Group-Wise Scheme. The scheme presented in Section 3.2 can be implemented in a fully distributed manner and the detection delay scales with ARL in a log order. However, compared with the case in which there is no dishonest sensor or the case in which the identity of the dishonest sensor is known, the pre-log factor of the scheme in Section 3.2 is larger, which implies that the delay increases faster with the ARL constraint. In particular, if there is no dishonest sensor, from (3.3), we know the pre-log factor is \(2/(N\mu^2)\). If we know the identity of the dishonest sensor, then the problem is reduced to a case with \(N-1\) honest sensors. Again from (3.3), we know that the pre-log factor is \(2/((N-1)\mu^2)\). On the other hand, the pre-log factor of the fully distributed scheme discussed in Section 3.2 is \(4/\mu^2\). In this section, we propose a modified scheme that achieves a better slope that decreases with increasing \(N\).

In the modified scheme, we evenly divide these \(N\) sensors into 3 groups, each with \(N/3\) sensors. Here we assume that \(N\) can be divided by 3. The scheme can be easily modified if \(N\) cannot be divided by 3. We use \(G_i, i=1,2,3\) to denote each group of sensors. For each group \(G_i\), we run the following scheme:

\[
T_{G_i}^h = \inf\{t \geq 0; y_{G_i}^t \geq h\},
\]

in which

\[
y_{G_i}^t = u_{G_i}^t - m_{G_i}^t
\]

with

\[
u_{G_i}^t = \sum_{n \in G_i} \mu_{n}^{(n)} - \frac{1}{3} N \mu^2 t, \quad \text{and} \quad m_{G_i}^t = \inf_{0 \leq s \leq t} u_{G_i}^s.
\]

In our scheme, the fusion center will raise an alarm when at least two out of these three groups raise alarms. In other words, \(T_h^G\), the time when the fusion center raises an alarm, is \(T_{(2,3)}^G\); the second order statistic of three random variables \(T_{G_1}^h, T_{G_2}^h, T_{G_3}^h\).

Now, we analyze the detection delay \(d(T_h^G)\) and ARL to false alarm \(\ARL(T_h^G)\). We have the following result:

**Theorem 3.6.** Under any attack strategy the detection delay \(T_h^G\) scales logarithmically with ARL in the following manner:

\[
d(T_h^G) \leq \frac{12}{\mu^2 N} \left( \log(\ARL(T_h^G)) + \log \frac{N\mu^2}{3} - 1 + o(1) \right).
\]

**Proof.** The proof follows that of Theorem 3.2 with only minor modification. In particular, in the group-wise scheme, \(u_{G_i}^t\) has a drift of \(-\mu^2 N/6 \) before the change, \(\mu^2 N/6 \) after the change, and its standard deviation is \(\mu \sqrt{N/3}\). Following the steps in the proof of Theorem 3.2, we obtain the estimate

\[
\ARL(T_h^G) \geq E_\infty\{T_{(1,2)}^G\} = \frac{3}{\mu^2 N} \left[ e^h + \text{“lower exponents”} \right].
\]

Similarly,

\[
d(T_h^G) \leq E_0\{T_{(2,3)}^G\} \leq 2E_0\{T_{h}^{G_i}\} = \frac{2}{\mu^2 N/6} (e^{-h} + h - 1).
\]

The statement of the theorem easily follows from the above two estimates. \(\square\)
Remark 3.7. This modified scheme achieves a smaller detection delay than the one discussed in Section 3.2. However, this modified scheme is a centralized scheme as the fusion center needs signals from all sensor in each group.

Remark 3.8. One may wonder whether we can further divide the sensors into two groups instead of three. There are some potential issues with dividing the sensors into two groups. If one employs the second-alarm scheme, the attacker can make the detection delay arbitrarily long by generating a Brownian motion with a sufficiently small negative drift. In this way, the combined signal in the group containing the attacker will not raise an alarm (or will raise an alarm very late) even if there is a true change in the environment. If, instead of using the second-alarm scheme, we opt to use the usual CUSUM scheme of Section 3.1, then the attacker can make ARL to false alarm very short by generating a Brownian motion with a sufficiently large positive drift.

4. Discrete-time Model

In this section, we discuss the discrete-time model for quickest change detection with compromised sensors. We follow the same structure as in the Brownian motion model.

4.1. The CUSUM Test. Similar to the Brownian motion case, when all nodes are honest, the non-Bayesian quickest change detection problem under Lorden’s performance index has been well understood. The optimal solution is Page’s CUSUM stopping rule [12]. In particular, define

\[ y_k = u_k - m_k, \]

with

\[ u_k = \sum_{l=1}^{k} \sum_{n=1}^{N} \log \left( \frac{f_1(\xi_l^{(n)})}{f_0(\xi_l^{(n)})} \right) \triangleq \sum_{l=1}^{k} \sum_{n=1}^{N} Z_l^{(n)}, \]

\[ m_k = \min_{0 \leq j \leq k} u_j. \]

Here, \( k \) is the time index. The CUSUM stopping rule is

\[ T_\nu = \inf\{k \geq 0; y_k \geq \nu\} \]

where \( \nu \) is a threshold such that

\[ E_1[T_\nu] = \gamma. \]

The performance evaluation of the discrete-time CUSUM is more involved than that of the Brownian motion model, mainly because of the possibility of overshoot. Define

\[ \kappa = \lim_{\nu \to \infty} E_1\{y_{T_\nu} - \nu\}, \]

\[ \beta = E_1\{m_\infty\}, \]

\[ R = \lim_{\nu \to \infty} E_1\{\exp[-u_{\eta_\nu} - \nu]\}, \]

with

\[ \eta_\nu = \inf\{k : u_k \geq \nu\}. \]
Here, $E_1$ denote the average under pdf $\prod_{i=1}^N f_i$.

From [19] one has

$$\text{ARL}(T_\nu) = E_\infty[T_\nu] = \frac{1}{R^2 N D(f_1||f_0)} e^\nu[1 + o(1)],$$

in which $D(f_1||f_0)$ is the Kullback-Leibler distance between $f_1$ and $f_0$. Here, $E_\infty$ denote the average under pdf $\prod_{i=1}^N f_i$.

In addition, from [5], one has

$$d(T_\nu) = E_1[T_\nu] = \frac{1}{ND(f_1||f_0)} (\nu + \beta_1 + \kappa_1) + o(1).$$

### 4.2. A Fully Distributed Byzantine Attack Resistant Change Detection Scheme

Here, we show that a scheme modified from the one discussed in Section 3.2 enables us to recover the logarithmic scaling between the detection delay and average run length. In this scheme, the fusion center first runs $N$ independent CUSUM $T_h^{(n)}$, one for each signal $\bar{\xi}^{(n)}_{k}$ received from sensor $n$:

$$T_h^{(n)} = \inf\{k \geq 0; y_h^{(n)}(k) \geq h\},$$

in which

$$y_h^{(n)}(k) = u_h^{(n)}(k) - m_h^{(n)},$$

with

$$u_h^{(n)}(k) = \sum_{l=1}^k \log \left\{ \frac{f_1(\bar{\xi}^{(n)}_l)}{f_0(\bar{\xi}^{(n)}_l)} \right\} \triangleq \sum_{l=1}^k Z_h^{(n)}_l,$$

$$m_h^{(n)} = \min_{0 \leq j \leq k} u_h^{(n)}(j).$$

Similar to the Brownian motion model, the fusion center will raise the final alarm when two alarms are raised from these $N$ CUSUMs. That is, $T_h$, the time the fusion center raises the final alarm, is the same as $T^{(2)}_h$, the second order statistics of $N$ random variables $T_h^{(1)}, \cdots, T_h^{(N)}$.

Now, we analyze the detection delay $d(T_h)$ and ARL to false alarm, which we denote by $\text{ARL}(T_h)$.

**Theorem 4.1.** Under any attack strategy, the detection delay scales logarithmically with $\text{ARL}$:

$$d(T_h) \leq \frac{2}{D(f_1||f_0)} \left( \text{log } \text{ARL}(T_h) + \log \left( \frac{1 - \exp(1 - N))R^2 D(f_1||f_0)}{\exp(1 - N)} + \beta_1 + \kappa_1 \right) + o(1).$$

**Proof.** The proof follows a structure similar to the Brownian motion case. We first provide a lower bound for ARL to false alarm for any strategy the attacker might use. From the attacker’s perspective, to reduce the ARL to false alarm of the proposed scheme, it should make the compromised sensor to raise an alarm as soon as possible. The proposed scheme will then raise a false alarm when one of the remaining $N - 1$ sensors raises an alarm under the probability measure $P_\infty$. Since all sensors are identical, for the evaluation of the performance purpose, we can simply assume sensor $N$ is compromised. Again, let $T^{(1)}_{(1),N-1} \triangleq \min\{T_h^{(1)}, \cdots, T_h^{(N-1)}\}$ be the first order statistics of
From the discussion above, we have $\text{ARL}(T_h) \geq E_\infty \{T_{(1),N-1}\}$. Since $T_{(1),N-1}$ is a positive random variable, we have

\begin{equation}
\text{ARL}(T_h) \geq E_\infty \{T_{(1),N-1}\} = \sum_{k=1}^\infty P_\infty (T_{(1),N-1} \geq k) = \sum_{k=1}^\infty P_\infty \left( \min \left\{ T_h^{(1)}, \ldots, T_h^{(N-1)} \right\} \geq k \right)
= \sum_{k=1}^\infty \left[ P_\infty (T_h^{(1)} \geq k) \right]^{N-1}.
\end{equation}

The last equality is due to the fact that for those sensors not affected by the attacker, $T_h^{(n)}$ are i.i.d. random variables.

Now, we provide an upper-bound for the detection delay under any attack. Clearly, to increase the detection delay of the proposed scheme, the attacker should use a strategy that does not raise an alarm. In this case, the fusion center will raise an alarm only when at least two honest sensors raise alarms under the probability measure $P_0$. Since all sensors are identical, for the evaluation of the performance purpose, we can simply assume sensor $N$ is compromised.

Let $T_{(2),N-1}$ be the second order statistics of $T_h^{(1)}, \ldots, T_h^{(N-1)}$. Based on the discussion above, we have

\begin{align*}
d(T_h) &\leq E_1 \{T_{(2),N-1}\} \\
&= \sum_{k=1}^\infty P_1 (T_{(2),N-1} \geq k) \\
&= \sum_{k=1}^\infty P_1 \left\{ \text{at most 1 of } T_h^{(1)}, \ldots, T_h^{(N-1)} \text{ is less than } k \right\} \\
&= \sum_{k=1}^\infty P_1 \left\{ \text{none of } T_h^{(1)}, \ldots, T_h^{(N-1)} \text{ is less than } k \right\} \\
&\quad + \sum_{k=1}^\infty P_1 \left\{ 1 \text{ of } T_h^{(1)}, \ldots, T_h^{(N-1)} \text{ is less than } k \right\} \\
&= \sum_{k=1}^\infty P_1 \left( \min \left\{ T_h^{(1)}, \ldots, T_h^{(N-1)} \right\} \geq k \right) + \sum_{k=1}^\infty P_1 \left\{ 1 \text{ of } T_h^{(1)}, \ldots, T_h^{(N-1)} \text{ is less than } k \right\} \\
&= \sum_{k=1}^\infty \left[ P_1 (T_h^{(1)} \geq k) \right]^{N-1} + \sum_{k=1}^\infty (N-1)(1-P_1 (T_h^{(1)} \geq k)) \left[ P_1 (T_h^{(1)} \geq k) \right]^{N-2}.
\end{align*}

Unlike the Brownian motion case, closed form expressions for $P_1 (T_h^{(1)} \geq k)$ and $P_\infty (T_h^{(1)} \geq k)$ are unknown. However, it has been shown in [6] that a properly normalized $T_h^{(1)}$ has a geometric distribution under $P_\infty$. In particular, letting

\begin{equation}
C = \frac{1}{R_1^2 D(f_1||f_0)} e^h[1 + o(1)],
\end{equation}


we have
\[
\lim_{h \to \infty} P_\infty \left\{ \frac{T_h^{(1)}}{C} \geq k \right\} = \exp(-k).
\]
Here, \( R_1 \) is the quantity corresponds to (4.5) when there is only one observation sequence.

We continue the computation.

\[
T_{(1),N-1} = \min \left\{ T_h^{(1)}, \cdots, T_h^{(N-1)} \right\} = C \min \left\{ T_h^{(1)}/C, \cdots, T_h^{(N-1)}/C \right\}.
\]

Following (4.7), we have

\[
\text{ARL}(T_h) \geq E_\infty \{ T_{(1),N-1} \} = C \sum_{k=1}^{\infty} \left[ P_\infty \left( T_h^{(1)}/C \geq k \right) \right]^{N-1}
\]

\[
= C \sum_{k=1}^{\infty} \exp(-(N-1)k)
\]

\[
= \frac{\exp(1-N)}{1-\exp(1-N)} C
\]

\[
= \frac{\exp(1-N)}{(1-\exp(1-N)) R_1^2 D(f_1||f_0)} e^{h[1+o(1)]}.
\]

Now, for \( d(T_h) \), as the Brownian motion case, \( P_1(T_{(2),N-1} \geq k) \) is a non-increasing function of \( N \). Hence from (4.2), we have

\[
d(T_h) \leq E_1 \{ T_{(2),N-1} \}
\]

\[
= \sum_{k=1}^{\infty} \left[ P_1(T_h^{(1)} \geq k) \right]^{N-1} + \sum_{k=1}^{\infty} (N-1)(1-P_1(T_h^{(1)} \geq k)) \left[ P_1(T_h^{(1)} \geq k) \right]^{N-2}
\]

\[
\leq \sum_{k=1}^{\infty} \left[ P_1(T_h^{(1)} \geq k) \right]^2 + \sum_{k=1}^{\infty} 2(1-P_1(T_h^{(1)} \geq k)) P_0(T_h^{(1)} \geq k)
\]

\[
= \sum_{k=1}^{\infty} 2 P_1(T_h^{(1)} \geq k) - \sum_{k=1}^{\infty} P_1^2(T_h^{(1)} \geq k)
\]

\[
= \frac{2}{D(f_1||f_0)} (h+\beta_1+\kappa_1) + o(1) - \sum_{k=1}^{\infty} P_1^2(T_h^{(1)} \geq k)
\]

\[
\leq \frac{2}{D(f_1||f_0)} (h+\beta_1+\kappa_1) + o(1),
\]

in which \( \kappa_1 \) and \( \beta_1 \) are the quantities corresponding to (4.3) and (4.4) when there is only one observation sequence.

As a result, we see that \( d\{ T_h \} \) grows logarithmically with \( \text{ARL}(T_h) \).

\[ \square \]

### 4.3. Group-Wise Scheme.

Similar to the Brownian motion case, we can design a group-wise scheme that has a smaller detection delay. In particular, we evenly divide these \( N \) sensors into 3 groups, each with \( N/3 \) sensors. For each group \( \mathcal{G}_i \), we run the following scheme:

\[
T_h^{\mathcal{G}_i} = \inf \{ k \geq 0; y_k^{\mathcal{G}_i} \geq h \},
\]
in which
\[ y^G_k = u^G_k - m^G_k \]
with
\[ u^G_k = \sum_{l=1}^{k} \sum_{n \in G_i} \log \left\{ \frac{f_1(\xi^{(n)}_l)}{f_0(\xi^{(n)}_l)} \right\}, \quad m^G_k = \min_{0 \leq j \leq k} u^G_j. \]

In our scheme, the fusion center will raise an alarm when at least two out of these three groups raise alarms. In other words, \( T^G_{h} \), the time when the fusion center raises an alarm, is \( T^G_{(2),3} \), the second order statistic of three random variables \( T^G_1, T^G_2, T^G_3 \). Regarding this group-wise scheme, we have the following result.

**Theorem 4.2.** For the group-wise scheme, the detection delay scales with ARL to false alarm in the following manner:
\[
d(T^G_h) \leq \frac{2}{D(f_1||f_0)N/3} \left( \log \text{ARL}(T^G_h) + \log \frac{(1 - \exp(-1))R^2_G D(f_1||f_0)N/3}{\exp(-1)} + \beta_G + \kappa_G \right) + o(1),
\]
in which \( \beta_G, \kappa_G \) and \( R_G \) are quantities corresponding to (4.3), (4.4) and (4.5) when there are \( N/3 \) sequences.

**Proof.** The proof follows closely with that of Theorem 4.1, with the difference that when all sensors in \( G_1 \) are honest, for \( T^G_1 \), we have
\[
d(T^G_1) = \frac{1}{D(f_1||f_0)N/3} (h + \beta_G + \kappa_G) + o(1),
\]
and \( T^G_1/C_G \) is geometrically distributed under \( P_\infty \), in which
\[
C_G = \frac{1}{R^2_G D(f_1||f_0)N/3} e^h [1 + o(1)],
\]
and
\[
\lim_{h \to \infty} P_\infty \left\{ \frac{T^G_1}{C_G} \geq k \right\} = \exp(-k).
\]

Following the same steps in the proof of Theorem 4.1, we obtain
\[
\text{ARL}(T^G_h) \geq \frac{\exp(-1)}{(1 - \exp(-1))R^2_G D(f_1||f_0)N/3} e^h [1 + o(1)].
\]
and
\[
d(T^G_h) \leq \frac{2}{D(f_1||f_0)N/3} (h + \beta_G + \kappa_G) + o(1).
\]
\[ \square \]
References

[1] M. Beibel, A note on Ritov’s Bayes approach to the minimax property of the CUSUM procedure, Annals of Statistics, 24 (1996), pp. 1804–1812.

[2] J. Chamberland and V. V. Veeravalli, Asymptotic results for decentralized detection in power constrained wireless sensor networks, IEEE Journal on Selected Areas in Communications, 22 (2004), pp. 1007–1015.

[3] R. W. Crow and S. C. Schwartz, Quickest detection for sequential decentralized decision systems, IEEE Trans. Aerosp. Electron. Syst., 32 (1996), pp. 267–283.

[4] H. A. David and H. N. Nagaraja, Order Statistics, Wiley, Hoboken, NJ, 2003.

[5] O. Hadjiliadis, H. Zhang, and H. V. Poor, One shot schemes for decentralized quickest change detection, IEEE Trans. Inform. Theory, 55 (2009), pp. 3346–3359.

[6] R. A. Khan, Detecting changes in probabilities of a multi-component process, Sequential Analysis, 14 (1995), pp. 375–388.

[7] L. Lai, Y. Fan, and H. Vincent Poor, Quickest detection in cognitive radio: A sequential change detection framework, in Proc. IEEE Global Telecommunications Conf., New Orleans, LA, 2008.

[8] L. Lamport, R. Shostak, and M. Pease, The Byzantine generals problem, ACM Transactions on Programming Languages and Systems, 4 (1982), pp. 382–401.

[9] G. Lorden, Procedures for reacting to a change in distribution, Annals of Mathematical Statistics, 42 (1971), pp. 1897–1908.

[10] M. Magdon-Ismail, A. F. Atiya, A. Pratap, and Y. S. Abu-Mostafa, On the maximum drawdown of Brownian motion, Journal of Applied Probability, 41 (2004), pp. 147–161.

[11] Y. Mei, Information bounds and quickest change detection in decentralized decision systems, IEEE Trans. Inform. Theory, 51 (2005), pp. 2669–2681.

[12] G. Moustakides, Optimal stopping times for detecting changes in distributions, Annals of Statistics, 14 (1986), pp. 1379–1387.

[13] ———, Optimality of the CUSUM procedure in continuous time, Annals of Statistics, 32 (2004), pp. 302–315.

[14] G. V. Moustakides, Decentralized CUSUM change detection, in Proc. International Conf. on Information Fusion, vol. 2, Nov. 2000, pp. 1089–1093.

[15] H. V. Poor and O. Hadjiliadis, Quickest Detection, Cambridge University Press, Cambridge, UK, 2008.

[16] S. W. Roberts, A comparison of some control chart procedures, Technometrics, 8 (1966), pp. 411–430.

[17] A. N. Shiryaev, Minimax optimality of the method of cumulative sums (CUSUM) in the continuous case, Russian Mathematical Surveys, 51 (1996), pp. 750–751.

[18] A. Tartakovsky, A. Polunchenko, and G. Sokolov, Efficient computer network anomaly detection by changepoint detection methods, IEEE Journal on Special Topics in Signal Processing, 7 (2013), pp. 4–11.

[19] A. G. Tartakovsky, Asymptotic performance of a multichart CUSUM test under false alarm probability constraint, in Proc. IEEE Conf. on Decision and Control, Seville, Spain, Dec. 2005, pp. 320–325.

[20] A. G. Tartakovsky and H. Kim, Performance of certain decentralized distributed change detection procedures, in Proc. Intl. Conf. on Information Fusion, Florence, Italy, July 2006, pp. 1–8.

[21] A. G. Tartakovsky, B. L. Rozovshki, R. B. Blazek, and H. Kim, A novel approach to detection of intrusions in computer networks via adaptive sequential and batch-sequential changepoint detection methods, IEEE Trans. Signal Processing, 54 (2006), pp. 3372–3382.

[22] A. G. Tartakovsky and V. V. Veeravalli, Quickest change detection in distributed sensor systems, in Proc. Intl. Conf. on Information Fusion, Cairns, Australia, July 2003, pp. 756–763.

[23] ———, Change-point detection in multichannel and distributed systems with applications, Marcel Dekker, New York, 2004.

[24] ———, Asymptotically optimal quickest change detection in distributed sensor systems, Sequential Analysis, 27 (2008), pp. 441–475.
[25] D. Teneketzis and Y. C. Ho, The decentralized quickest detection problem, IEEE Trans. Automatic Control, 29 (1984), pp. 641–644.

[26] V. V. Veeravalli, Decentralized quickest change detection, IEEE Trans. Inform. Theory, 47 (2001), pp. 1657–1665.

[27] P. Willett, P. F. Swaszek, and R. S. Blum, The good, bad, and ugly: Distributed detection of a known signal in dependent Gaussian noise, IEEE Trans. Signal Processing, 48 (2000), pp. 3266–3279.

(Erhan Bayraktar) Department of Mathematics, University of Michigan, 530 Church Street, Ann Arbor, MI 48104, USA

E-mail address: erhan@umich.edu

(Lifeng Lai) Department of Electrical and Computer Engineering, Worcester Polytechnic Institute, Worcester, MA 01609, USA

E-mail address: llai@wpi.edu