This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the ICC 2007 proceedings.

A Robust Statistical Scheme to Monitor Transient Phenomenon in Sensor Networks

Vinod Shukla and Daji Qiao
Iowa State University, Ames, IA 50011
{vkshukla, daji}@iastate.edu

Abstract— **Wireless sensor networks have been deployed for various critical monitoring applications in hostile environments such as monitoring the concentration levels of hazardous gas species in a battle field. Due to the sensitivity of such applications, it becomes mandatory to record the transient variations in the phenomenon, and take corrective actions, if necessary. At the same time it is important to shield the network from false data injection by adversaries who intend to disrupt the functioning of the system. While many schemes exist to prevent false data injection, they are counterproductive to preserving the transient observations. We devise a robust statistical scheme to monitor transient phenomenon while being immune to false data injection attacks. The key idea of our scheme is to require each sensor node to report a statistical digest of recent sensed readings in addition to the current reading; then inter-sensor statistical tests are designed and utilized to help preserve transient data while restricting the impact of false data injection significantly. Detailed theoretical analysis and in-depth simulations are presented to corroborate our scheme.**

I. Introduction

Application of wireless sensors for monitoring applications has gathered significant attention from research community recently. Numerous sensors are deployed to monitor a particular area which could range from an industrial setup to an environmental habitat. The network is responsible for collecting data and sending it to a distant Base Station (BS) for monitoring and taking requisite actions, if any. Due to large number of sensors, high volume of data is generated and the network is typically organized into clusters, with a Cluster Head (CH) that is responsible for local decision making and aggregation of data to be forwarded to BS.

The data being reported is prone to all sorts of malicious attacks, where the adversary can compromise a sensor node and gain access to all the stored information. It can also alter the contents of the data in order to make BS accept a false value. Such an attack is called False Injection attack [1]. Typically, this attack has been addressed using schemes [1]–[5] wherein the sensors report only the readings and a certain designed number of sensors should agree (similar to majority voting) for the reading to be considered valid. Such a scheme has an inherent limitation: in the presence of a transient phenomenon, the data source will be varying and sensors may not agree with each other even though all reported data is genuine; hence, data reported by a sensor may be classified wrongly as false and rejected. Moreover, a monitoring application would be typically interested in recording the transient phenomenon and taking necessary actions. Hence, we devise a scheme which preserves the transient variation while being simultaneously immune to false injection attacks.

The problem of distinguishing transient data from false data has not been studied in great depth before. Various secure aggregation schemes [6]–[9] designed to compute specific aggregates don't find direct application in our goal of distinguishing data transience from false injection. In a recent work [10], we propose SSTF, a novel scheme for distinguishing data transience from false injection. SSTF employs a statistical framework on top of a suitably enhanced version of an existing security scheme to achieve its goals. In contrast to [10], in this paper we concentrate on the statistical part only and present theoretical analysis to demonstrate the robustness of our scheme. We derive maximum possible false injection on the reported reading that would be accepted in our scheme, which shows that a compromised node will never be able to make a significant impact in our scheme regardless of the false readings it may report.

The rest of the paper is organized as follows. We give the system model and problem statement in Section II. We describe the proposed scheme in Section III and give a detailed security analysis in Section IV. Simulation results are presented in Section V and we conclude the paper by presenting conclusions and future work in Section VI.

II. Models and Problem Statement

A. System Model

The wireless sensor network is partitioned into clusters; each cluster has a Cluster Head (CH) and a set of sensor nodes, which gather information and transmit it to CH. CH does decision making and aggregation on the information received from the sensors and forwards an aggregated report to a distant Base Station (BS).The sampling rate of sensors depends on the maximum phenomenon variation per unit time as well as the spatial diffusion rate. Instead of sending only the sensed reading to CH, each sensor does a local computation over a sliding window of recent sensed readings and sends a statistical digest to CH at periodic intervals.

B. Threat Model

Sensor nodes may be compromised or physically captured. All the secret information stored in a compromised node can be accessed by the adversary. Various attacks like dropping reports or altering the message contents can be launched, so as to prevent the base station from receiving authentic sensor readings. Also, there may be colluded attacks where two or

1-4244-0353-7/07/$25.00 ©2007 IEEE

more nodes collaborate to let the false reports escape detection in the downstream path to the BS.

C. Problem Statement

If the phenomenon being sensed observes transient variations, instantaneous sensor readings recorded by individual sensors in a cluster may vary. Even though other sensors do not immediately agree, they should sense similar transient variation after a few samples due to the diffusing nature of the phenomenon. Also, such transient data is genuine and should be preserved. A compromised node (or group of colluding compromised nodes) will try to inject a false reading into the network and our aim is to minimize the impact of false injection on the aggregation process and detect it eventually. Thus, we identify the following design goals for our scheme:

- It should distinguish genuine transient data from injected false data and report them with low false positives;
- False data injection should have minimal impact on the aggregation process and be detected as soon as possible.

III. Proposed Scheme

We propose a robust statistical scheme to monitor transient phenomenon in wireless sensor networks while being immune to false injection attacks. Our scheme has four aspects of operation: *Individual Sensor Behavior; CH Behavior; Sensor Endorsement;* and *En-route Nodes and BS Behavior*.

Security Assumptions: Similar to others [1], [10], we proceed with the following security assumptions.Every node shares a master secret key with BS. Each node knows its one-hop neighbors and has established a pairwise key with each of them. A node can establish a pairwise key with another node that is multiple hops away, if needed. All the nodes are equally trusted and if a node is compromised, all the information it holds will also be compromised. However, it is assumed that BS can not be compromised. We consider a clustered wireless sensor network and there can be either one-to-one correspondence or many-to-one correspondence established between the cluster nodes and the en-route nodes to BS. With a proper association scheme and an en-route filtering scheme, it is ensured that as long as a valid cluster node does not sign the false aggregated report, it will eventually be detected en-route and dropped before being propagated further. In the rest of this paper, aforementioned security assumptions hold, and more details of the security scheme can be found in [10] and [1]. Table I summarizes the notations used in this paper.

TABLE I

Notations used in the paper

Notation	Remarks
$\mathcal{P}$	Phenomenon being sensed by a cluster.
$\mathcal{D}$	Diffusion rate of $\mathcal{P}$, measured in units/sec.
ρ	Phenomenon variation rate: maximum change in the phenomenon per unit time measured in units/sec (e.g. ppm/sec for gas concentration, etc).
x	Sampling rate at each sensor, measured in samples/sec.
d	Maximum inter-node distance between any two sensor nodes within a cluster, measured in meters.
n	Reporting interval: each sensor sends report to CH every n samples.
w	Size of the sliding window.
τ	Number of nodes in the cluster (including CH).
v_k	A sensor in the cluster other than CH.
r_k	Sensed reading reported by v_k in a report.
μ_k	Sample mean reported by v_k in a report.
σ_k^2	Sample variance reported by v_k in a report.
R_k	A report sent by v_k in the format of (r_k, μ_k, σ_k^2).
R_{Ag}	The aggregated report generated by CH in the format of $(r_{Ag}, \mu_{Ag}, \sigma_{Ag}^2)$.

A. Individual Sensor Behavior

A sliding window implementation is instituted at each individual sensor node in the cluster. A sensor node senses the phenomenon at the sampling rate. It maintains a buffer size equal to that of the sliding window (w) to store the w most recent readings. Every time a new reading is sensed, the oldest one is deleted, thus maintaining a sliding window of size w at each sensor. We need to have w samples to generate a report. After every reporting interval (n samples), the sensor node v_k computes the sample mean (μ_k), and sample variance (σ_k^2) over the sliding window (w samples). This is further illustrated in Fig. 1. The report from sensor node v_k to CH is in the format of $R_k \equiv (r_k, \mu_k, \sigma_k^2)$.

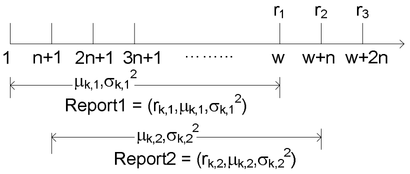


Fig. 1. Sliding window implementation and report generation at sensor node v_k. r, μ, σ are respectively the last reading, mean and standard deviation of w samples in a sliding window. Shown are reports for two windows (i = 1, 2) at v_k. There are n non-overlapping samples between two adjacent windows.

B. Cluster Head Behavior

In addition to performing the same functions as other sensors in the cluster, CH collects the reports R_k from all individual sensors for testing and aggregation.

CH performs three inter-sensor tests. First, CH verifies the bounds on the readings reported by individual sensors, and if true, CH does a pairwise inter-sensor test to verify conformity of the reported distributions. Finally, CH does a *bin test* on the reports that pass the previous two tests to further limit the impact of false data by utilizing the reported distributions.

1) Bound Test: Bound test checks for the maximum possible difference between the readings reported by the sensors. The maximum inter-node distance is d and phenomenon diffusion rate is $\mathcal{D}$. So the maximum time taken for a reading to diffuse between two sensors is $\frac{d}{\mathcal{D}}$. Since the phenomenon can change at a maximum rate of Phenomenon variation rate ρ, the maximum difference between readings reported by any two sensors is $\rho\frac{d}{\mathcal{D}}$. Thus the readings reported by two well-behaving sensors j and k should satisfy:

$$|r_j - r_k| \leqslant \rho\frac{d}{\mathcal{D}}.$$

2) Distribution Test: For the sensors that have passed the bound test, CH does a further pairwise test to check whether the distributions reported by the sensors conform to each other. CH takes the means μ_k reported by the sensors as measurements of a common mean. For two sensors v_i and v_j, CH does a z-test to check whether the means μ_i and μ_j are equal with α confidence level, where α is a design parameter and the desired α can be achieved by adjusting the sliding window size. The z-test procedure is described in Fig. 2.

z-test for equality of two sensor reports

For any two sensors v_i**,** v_j:
$\mathcal{N}(\mu_i, \sigma_i^2)$: Distribution reported by v_i
$\mathcal{N}(\mu_j, \sigma_j^2)$: Distribution reported by v_j
To test $\mu_i = \mu_j$:

1) Compute Standard deviation of the difference distribution: $\sigma_{ij}^2 = \sigma_i^2 + \sigma_j^2$.
2) Compute Standard error of the means: $z_{ij} = \frac{|\mu_i - \mu_j|}{\sqrt{\frac{\sigma_{ij}^2}{w}}}$.
3) The condition for $\mu_i = \mu_j$ with α confidence level is $z_{ij} \leqslant z_\alpha \times \sigma_{ij}$. In this paper we use confidence level of $\alpha = 90\%$; correspondingly $z_\alpha = 0.1257$. Substituting for z_{ij} and σ_{ij}, the condition becomes: $|\mu_i - \mu_j| \leqslant z_\alpha \times \frac{\sigma_i^2 + \sigma_j^2}{\sqrt{w}}$.

Fig. 2. Procedure for z-test

For the sensors that pass the distribution test, CH proceeds to calculate the aggregated mean and variance based on the sample means and variances reported by the individual sensor nodes. Specifically, CH takes the means reported by individual sensors as measurements of a common aggregated mean which needs to be computed. Under this assumption, the aggregated mean and variance can be computed by using Maximum Likelihood Estimation (MLE):

$$\begin{cases} \mu_{Ag} = \frac{\sum_{k=1}^{\gamma} \mu_k / \sigma_k^2}{\sum_{k=1}^{\gamma} 1/\sigma_k^2}, \\ \sigma_{Ag}^2 = (\sum_{k=1}^{\gamma} 1/\sigma_k^2)^{-1}, \end{cases} \tag{1}$$

where $\gamma \leqslant \tau$ is the number of sensors that passed the distribution test. τ is the total number of nodes in the cluster.

3) Bin Test: The bound test limits the false readings to be accepted only if they are within known bounds of variation, still this could result in a large error. We utilize the aggregated variance produced at the end of the distribution test to further limit the impact of false data on the aggregation process with the following bin test. Bin test is performed only on the readings reported by individual sensors that have passed both the bound test and the distribution test, called the *eligible sensors*. For each eligible sensor v_k, CH utilizes the aggregated variance to form a bin of size $[r_k - 2\sigma_{Ag}, r_k + 2\sigma_{Ag}]$. Then it checks if the reading reported by other eligible sensors lie in this bin. CH does this for every eligible sensor. Once it knows the bin size of all eligible nodes, it picks one with the largest size and averages the readings to compute a final aggregated reading r_{Ag}. This is illustrated in Fig. 3.

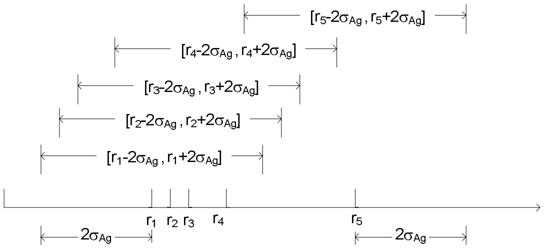


Fig. 3. Bin Test. Number of sensors in Bin1, Bin2, Bin3, Bin4, Bin5 is 4,4,4,4,1 respectively. Hence final $r_{Ag} = \frac{1}{4}(r_1 + r_2 + r_3 + r_4)$.

After r_{Ag} is computed, CH sends the aggregated report $R_{Ag} = (r_{Ag}, \mu_{Ag}, \sigma_{Ag}^2)$ to the individual sensors belonging to the selected bin for endorsement.

C. Sensor Endorsement

When a sensor v_k receives the aggregated report R_{Ag} from CH for endorsement, which is done to prevent CH from lying about the aggregation process, it performs the following tests:

- $\sigma_{Ag} \leqslant \sigma_k$;
- z-test to test whether $\mu_{Ag} = \mu_k$ as described in Fig. 2;
- bin test on r_{Ag}: $r_{Ag} \in [r_k - 2\sigma_{Ag}, r_k + 2\sigma_{Ag}]$.

v_k endorses R_{Ag} only if the above conditions are met. If v_k is able to endorse , it signs R_{Ag} using two keys, one it shares with an en-route node and one with BS [1], [10]. v_k sends this endorsed report to CH. Since CH does not have knowledge of any of these two keys, it can make no further changes to the endorsed reports.

D. En-route Nodes and Base Station Behavior

When CH receives endorsements from individual sensors, it merges them into a single report to be forwarded towards BS. When an en-route node receives the report, it verifies the integrity of the report by checking the endorsement key. If it is able to verify the entry, it forwards the report to the next en-route node, else it drops the report. The process thus continues to BS. If the verification at BS succeeds, the report R_{Ag} is accepted, else it is discarded. Interested readers may refer to [1], [10] for details of the en-route node filtering scheme, which is not the focus of this paper, and is omitted due to space limitations. BS records all the reports from each CH in the network, and uses them to depict the variations in the phenomenon.

E. Example

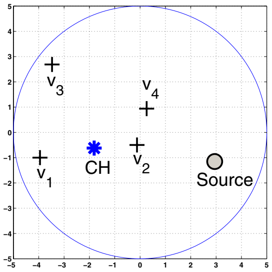


Fig. 4. An example to illustrate Bound Test, Distribution Test and Bin Test.

Consider a sensor cluster shown in Fig 4. Source, four sensor nodes and CH are randomly placed in a circle of radius 5 units. The source exhibits random variations in the source data as shown in Fig. 5. The window size $w = 100$ samples. Table II lists the reports generated by the sensors at a particular instant.

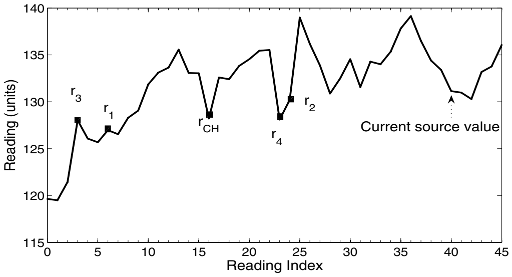


Fig. 5. Source data. Shown are the readings sensed by the sensors v_i; $i = 1, 2, 3, 4, CH$. There is a delay in the reading measured based on distance from the source.

TABLE II

LIST OF DATA REPORTS GENERATED BY SENSORS IN THE EXAMPLE

Sensor	r	μ	σ^2
v_1	127.719	100.396	735.982
v_2	130.709	99.320	734.46
v_3	127.680	100.572	735.06
v_4	127.761	99.441	734.41
CH	128.519	99.787	734.14

- Bound test:
  - $\rho \frac{d}{D} = 250$;
  - Since $\max_{i,j \in \{1,2,3,4,CH\}} |r_i - r_j| = 3.029 < 250$, all inter-sensor bound tests are passed.
- Distribution Test: Distribution test is performed only on μ_i and σ_i^2 reported by the sensors. For example, for sensors v_1 and v_2, $|\mu_1 - \mu_2| = 1.0759$ which is less than $(z_\alpha \times \frac{\sigma_1^2 + \sigma_2^2}{\sqrt{w}} = 18.484)$. It is verified that all the pairwise distribution tests pass. As a result, CH computes the aggregated mean $\mu_{Ag} = 99.9034$ and aggregated variance $\sigma_{Ag}^2 = 146.97$.
- Bin test: The bins are constructed around sensor readings. Since $\sigma_{Ag} = \sqrt{146.97} = 12.1231$, we have
  - Bin1: $\mu_1 + 2\sigma_{Ag} \equiv [103.4729, 151.9652]$;
  - Bin2: $\mu_2 + 2\sigma_{Ag} \equiv [106.4629, 154.9552]$;
  - Bin3: $\mu_3 + 2\sigma_{Ag} \equiv [103.4338, 151.9262]$;
  - Bin4: $\mu_4 + 2\sigma_{Ag} \equiv [103.5148, 152.0072]$;
  - BinCH: $\mu_{CH} + 2\sigma_{Ag} \equiv [104.2728, 152.7652]$.

  It can be seen that all the sensor readings belong to each of the bins. Thus the largest bin size is 5 and $r_{Ag} = (r_1 + r_2 + r_3 + r_4 + r_5)/5 = 128.4776$.
- Sensor Endorsement: CH sends the aggregated report $R_{Ag} \equiv (r_{Ag}, \mu_{Ag}, \sigma_{Ag}^2)$ to the the sensors for endorsement. When sensor v_1 receives R_{Ag}, it tests:
  - $\sigma_{Ag} \leqslant \sigma_1$: true since $12.1231 < 27.129$;
  - z-test to test $\mu_{Ag} = \mu_1$: true since $|\mu_{Ag} - \mu_1| = 0.4934$ which is less than $(z_\alpha \times \frac{\sigma_1^2 + \sigma_{Ag}^2}{\sqrt{w}} = 11.099)$;
  - bin test on r_{Ag}: true since $128.4776 \in [103.4729, 151.9652]$.

  Similarly, all other sensors do the same verifications before endorsing R_{Ag}.

IV. SECURITY ANALYSIS

A. *Effect of False Data Injection with the simple 1-step Bound Test*

Consider Fig. 6. r is a valid measurement. Let $\mathcal{B} = \rho \frac{d}{\mathcal{D}}$ denote the maximum possible difference between the readings allowed by the Bound Test. r_{min} and r_{max} are respectively the true minimum and maximum readings taken amongst all the sensors. Assume that the readings have a uniform distribution over $[r_{min}, r_{max}]$. Let $\mathcal{W}_r = r_{max} - r_{min}$ denote the width of this interval. The maximum possible reading that can escape the bound test is given by $r_{min} + \mathcal{W}_r$. The compromised node wants a false data $r^{'} = r + \Delta_r$ to get accepted.

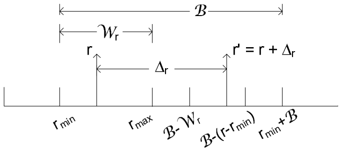


Fig. 6. Imposing limits based on the Bound Test.

We are interested in computing the maximum possible expected distortion that an attacker can inject without being detected i.e. we want to maximize the expectation E$[\Delta_r|\Delta_r$ is accepted]. For a given r, let $I_A(\Delta_r)$ denote the indicator function whether Δ is accepted, i.e.

$$I_A(\Delta_r) = \begin{cases} 1, & 0 \leqslant \Delta_r \leqslant r_{min} + \mathcal{B} - r, \\ 0, & \text{Otherwise.} \end{cases}$$

Then we have

E$[\Delta_r|\Delta_r$ is accepted]

$$\begin{aligned} &= \int_{r_{min}}^{r_{max}} I_A(\Delta_r)\Delta_r \frac{dr}{r_{max} - r_{min}} \\ &= \begin{cases} \Delta_r, & \Delta_r \leqslant \mathcal{B} - \mathcal{W}_r, \\ \frac{(\mathcal{B} - \Delta_r)\Delta}{\mathcal{W}_r}, & \mathcal{B} - \mathcal{W}_r < \Delta_r \leqslant \mathcal{B}, \\ 0, & \Delta > \mathcal{B}. \end{cases} \end{aligned} \tag{2}$$

Differentiating $E[\Delta_r]$ with respect to Δ_r, we get the optimal $\Delta_r = \Delta_r^*$ given by:

$$\Delta_r^* = \begin{cases} \mathcal{B} - \mathcal{W}_r, & \mathcal{W}_r < \frac{\mathcal{B}}{2}, \\ \frac{\mathcal{B}}{2}, & \mathcal{W}_r \geqslant \frac{\mathcal{B}}{2}. \end{cases} \tag{3}$$

Subsequently, the maximum expectation is given by:

$$\begin{aligned} E_{max} &= \max_{\Delta_r} \mathrm{E}[\Delta_r|\Delta_r \text{is accepted}] \\ &= \begin{cases} \mathcal{B} - \mathcal{W}_r, & \mathcal{W}_r < \frac{\mathcal{B}}{2}, \\ \frac{\mathcal{B}^2}{4\mathcal{W}_r}, & \mathcal{W}_r \geqslant \frac{\mathcal{B}}{2}. \end{cases} \end{aligned} \tag{4}$$

Fig. 7 illustrates the variation of expectation with respect to Δ_r. We can see that Δ_r is dependent on $\mathcal{B}$ and $\mathcal{W}_r$. When the source variation is less, $\mathcal{W}_r$ is small and the compromised

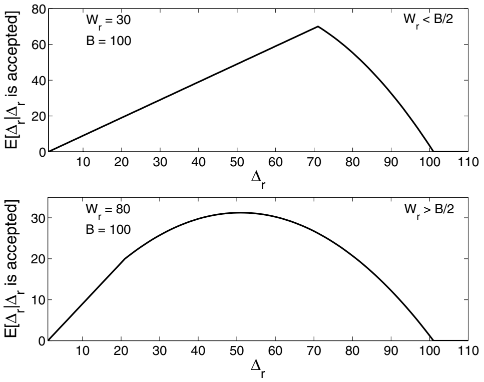


Fig. 7. E[$\Delta_r|\Delta_r$ is accepted] vs. Δ_r.

node should report $r^{'} = r + \mathcal{B} - \mathcal{W}_r$; in case of a highly varying source, $\mathcal{W}_r$ is large and the compromised node should report $r^{'} = r + \frac{\mathcal{B}}{2}$. Assuming there are $\mathcal{K}$ nodes participating in the aggregation process, on an average, a single compromised node is able to distort the aggregated reading r_{Ag} by $E_{max}/\mathcal{K}$, where E_{max} is given by Eq. (4). As such, with simple 1-step bound test, the network can still suffer pretty significant injection effects.

B. Effect of False Data Injection with the proposed 3-step testing scheme

1) Effect on μ_{Ag} and σ^2_{Ag}: Our primary measurement of interest are the readings and not the mean and variance. Nonetheless, mean and variance are important for the functioning of our scheme. In the following we show that reporting false mean or variance does not help the attacker.

A pairwise distribution test is performed to test the equality of means reported by the sensor nodes. Let v_j be a compromised node with true mean and variance of (μ, σ^2). Assume v_j reports $(\mu^{'}, \sigma^{'2})$ instead of the true values. To pass the distribution tests, the following conditions should hold:

$$|\mu^{'} - \mu_k| \leqslant z_\alpha \frac{\sigma^{'2} + \sigma_k^2}{\sqrt{w}}; \quad \forall k \in [1, \tau], k \neq j,$$

where (μ_k, σ_k^2) is the distribution reported by sensor v_k and τ is the number of nodes in the cluster.

μ_{min}, μ_{max} are the minimum and maximum means reported by the sensor nodes. All the reported means are assumed to be normally distributed over $[\mu_{min}, \mu_{max}]$. Let $\mathcal{W}_\mu = \mu_{max} - \mu_{min}$ denote the width of this interval. When the adversary plans to inject false mean and variance, it assumes all other sensors have the variance equal to its genuine variance. This is justified because the phenomenon is assumed to be a diffusion process and all sensors are sensing the same phenomenon. Also the computation is being done over a large sliding window, promoting homogeneity of variances.

Let $\mu^{'} = \mu + \Delta_\mu$ and $\sigma^{'2}$ be the false mean and variance reported by the compromised node. For a given Δ_μ, let $I_A(\sigma^{'2})$ denote the indicator function whether $\sigma^{'2}$ is accepted. I_A can be written separately for two cases as shown below:

- When $\Delta_\mu \geqslant \frac{\mu_{min}+\mu_{max}}{2}$:

$$I_{A1}(\sigma^{'2}) = \begin{cases} 1, & \mu + \Delta_\mu - \mu_{min} \geqslant \frac{z_\alpha}{\sqrt{w}}(\sigma^{'2} + \sigma^2), \\ 0, & \text{Otherwise.} \end{cases}$$

- When $\Delta_\mu < \frac{\mu_{min}+\mu_{max}}{2}$:

$$I_{A2}(\sigma^{'2}) = \begin{cases} 1, & \mu + \Delta_\mu - \mu_{min} \leqslant \frac{z_\alpha}{\sqrt{w}}(\sigma^{'2} + \sigma^2) \text{ \&} \\ & \mu + \Delta_\mu > \frac{\mu_{min}+\mu_{max}}{2}, \\ 1, & \mu_{max} - (\mu + \Delta_\mu) \leqslant \frac{z_\alpha}{\sqrt{w}}(\sigma^{'2} + \sigma^2) \text{ \&} \\ & \mu + \Delta_\mu < \frac{\mu_{min}+\mu_{max}}{2}, \\ 0, & \text{Otherwise.} \end{cases}$$

If the goal of the attacker is to minimize the aggregated standard deviation σ^2_{Ag}, so as to disturb the Bin Test and consequently the final aggregated reading, it would try to report the smallest $\sigma^{'2}$ that will be accepted. This will reduce σ^2_{Ag} and hence the bin size, such that the number of sensors constituting the largest bin reduces while the compromised node may still be a part of the largest bin. This can be obtained by maximizing the expectation

$$E\left[\tfrac{1}{\sigma^{'2}}|\sigma^{'2} \text{ is accepted}\right] = \int_\Omega I_A(\sigma^{'2}) \tfrac{1}{\sigma^{'2}} \tfrac{d\mu}{\mu_{max}-\mu_{min}}, \quad (5)$$

where Ω denotes the range of valid means. Similar to the indicator function, E[$\frac{1}{\sigma^{'2}}|\sigma^{'2}$ is accepted] can be computed for two cases as shown below:

- When $\Delta_\mu \geqslant \frac{\mu_{min}+\mu_{max}}{2}$:
  $E\left[\frac{1}{\sigma^{'2}}|\sigma^{'2} \text{ is accepted}\right] =$

$$\begin{cases} \frac{1}{\sigma^{'2}}, & \sigma^{'2} \geqslant \frac{(\mathcal{W}_\mu + \Delta_\mu)\sqrt{w}}{z_\alpha} - \sigma^2, \\ \frac{\frac{z_\alpha}{\sqrt{w}}(\sigma^2 + \sigma^{'2}) - \Delta_\mu}{\sigma^{'2}\mathcal{W}_\mu}, & \frac{\Delta_\mu \sqrt{w}}{z_\alpha} - \sigma^2 \leqslant \sigma^{'2} < \frac{(\mathcal{W}_\mu + \Delta_\mu)\sqrt{w}}{z_\alpha} - \sigma^2, \\ 0, & \sigma^{'2} < \frac{\Delta_\mu \sqrt{w}}{z_\alpha} - \sigma^2. \end{cases} \quad (6)$$

- When $\Delta_\mu < \frac{\mu_{min}+\mu_{max}}{2}$:
  $E\left[\frac{1}{\sigma^{'2}}|\sigma^{'2} \text{ is accepted}\right] =$

$$\begin{cases} \frac{1}{\sigma^{'2}}, & \sigma^{'2} \geqslant \frac{(\mathcal{W}_\mu + \Delta_\mu)\sqrt{w}}{z_\alpha} - \sigma^2, \\ \frac{\frac{z_\alpha}{\sqrt{w}}(\sigma^2 + \sigma^{'2}) - \Delta_\mu}{\sigma^{'2}\mathcal{W}_\mu}, & \frac{(\mathcal{W}_\mu - \Delta_\mu)\sqrt{w}}{z_\alpha} - \sigma^2 \leqslant \sigma^{'2} < \frac{(\mathcal{W}_\mu + \Delta_\mu)\sqrt{w}}{z_\alpha} - \sigma^2, \\ \frac{\frac{2z_\alpha}{\sqrt{w}}(\sigma^2 + \sigma^{'2}) - \mathcal{W}_\mu}{\sigma^{'2}\mathcal{W}_\mu}, & \frac{\mathcal{W}_\mu}{2}\frac{\sqrt{w}}{z_\alpha} - \sigma^2 \leqslant \sigma^{'2} < \frac{(\mathcal{W}_\mu - \Delta_\mu)\sqrt{w}}{z_\alpha} - \sigma^2, \\ 0, & \sigma^{'2} < \frac{\mathcal{W}_\mu}{2}\frac{\sqrt{w}}{z_\alpha} - \sigma^2. \end{cases} \quad (7)$$

From the above expressions, it can be seen that the expectation decreases as Δ_μ increases. Hence if the attacker wants to maximize the expectation, it should choose $\Delta_\mu = 0$. On the other hand, we can see that the expectation increases as $\sigma^{'2}$ increases up to $\sigma^{'2} = \frac{(\mathcal{W}_\mu + \Delta_\mu)\sqrt{w}}{z_\alpha} - \sigma^2$ and then it starts decreasing. Hence, the maximum expectation occurs when

$$\begin{cases} \Delta^*_\mu = 0, \\ \sigma^{*2} = max\left(\frac{\mathcal{W}_\mu \sqrt{w}}{z_\alpha} - \sigma^2, 0\right). \end{cases} \quad (8)$$

The maximum expectation is $E_{max} = \frac{1}{\left(\frac{\mathcal{W}_\mu \sqrt{w}}{z_\alpha} - \sigma^2\right)^2}$. The dependence of E_{max} on Δ_μ and $\sigma^{'2}$ is shown in Fig. 8. The plot is shown for the case when $\Delta_\mu \leqslant \frac{\mu_{min}+\mu_{max}}{2} (= 25)$. Plot for $\Delta_\mu > 25$ is not shown since it will have smaller maximum expectation and will decrease with increasing Δ_μ. We can see three distinct surfaces, which correspond to the three cases in Eq. (7).

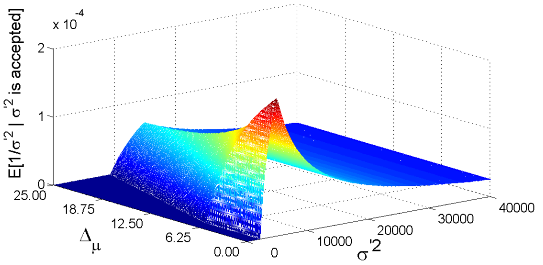


Fig. 8. $\mathrm{E}\Big[\frac{1}{\sigma^{'2}} | \sigma^{'2} \text{ is accepted}\Big]$ vs. Δ_μ and $\sigma^{'2}$.

Thus, compromised node should report a false mean $\mu^* = \mu + \Delta_\mu^* = \mu$ and a false variance $\sigma^{*2} = \frac{\mathcal{W}_\mu \sqrt{w}}{z_\alpha} - \sigma^2$. The effect on aggregated mean and variance is shown below:

$$\begin{cases} \mu_{Ag}^* = \dfrac{\sum_{\substack{k=1\\k\neq j}}^{\gamma} \mu_k/\sigma_k^2 + \mu/(\frac{\mathcal{W}_\mu \sqrt{w}}{z_\alpha} - \sigma^2)}{\sum_{\substack{k=1\\k\neq j}}^{n} 1/\sigma_k^2}, \\ \sigma_{Ag}^{*2} = \left(\sum_{\substack{k=1\\k\neq j}}^{\gamma} 1/\sigma_k^2 + 1/(\frac{\mathcal{W}_\mu \sqrt{w}}{z_\alpha} - \sigma^2)\right)^{-1}, \end{cases} \tag{9}$$

where γ is the number of sensors that passed distribution tests.

On the other hand, if goal of the attacker is to increase σ_{Ag}^2 so that it can report a higher $\mu^{'}$ to be accepted. It is noted from Eq. (1) that σ_{Ag}^2 will be the largest when $\sigma^{'2} = \infty$. Hence, the adversary reports $\sigma^{*2} = \infty$. Consequently, regardless of the $\mu^{'}$ being reported, the effect on μ_{Ag}^* and σ_{Ag}^{*2} is given by:

$$\begin{cases} \mu_{Ag}^* = \dfrac{\sum_{\substack{k=1\\k\neq j}}^{\gamma} \mu_k/\sigma_k^2}{\sum_{\substack{k=1\\k\neq j}}^{n} 1/\sigma_k^2}, \\ \sigma_{Ag}^{*2} = \left(\sum_{\substack{k=1\\k\neq j}}^{\gamma} 1/\sigma_k^2\right)^{-1}. \end{cases} \tag{10}$$

2) Effect on r_{Ag}: If only the simple 1-step bound test is implemented, an attacker injects false reports using Δ_r^* calculated in Eq. (3). In such a scenario, the attacker can succeed to induce a large error into the aggregated readings as the bounds are still very loose. However, in our scheme, due to the additional application of distribution test and bin test, the limits on the false data being accepted is tightly restricted.

After distribution test is passed, CH computes the aggregated mean (μ_{Ag}) and aggregated variance (σ_{Ag}^2). In the presence of compromised node, the computed aggregate mean and variance (μ_{Ag}^* and σ_{Ag}^{*2}) are bounded by Eqs. (9) and (10). CH then does a bin test on the eligible sensors and computes the aggregated reading (r_{Ag}) by averaging the reading of the sensors belonging to the largest bin. As such, if an attacker wants to report a false reading, it has to report a reading which will be included in the bin.

With these tests into effect, bound on readings reported by the attacker can be calculated as follows. Conventions are the same as in Section IV-A. r is a valid measurement. r_{min} and r_{max} are respectively the true minimum and maximum readings reported amongst the sensors. The reading r is assumed to a uniform distribution over $[r_{min}, r_{max}]$. The maximum reading that a compromised node can report to escape detection is $r_{min} + 2\sigma_{Ag}^*$. σ_{Ag}^{*2} is the optimal affected aggregated variance based on the optimal σ^{*2} reported by the compromised node. The goal of the attacker could have been to either increase or decrease σ_{Ag}^2 as discussed in Section IV-B.1.

The compromised node wants a false value $r^{'} = r + \Delta_r$ to get accepted. We are interested in computing maximum possible expected distortion that attacker can inject without being detected, which is obtained by maximizing the conditional expectation E[$\Delta_r | \Delta_r$ is accepted]. Following a similar analysis as in Section IV-A, replacing $\mathcal{B}$ by $2\sigma_{Ag}^*$, we can get the optimal Δ_r:

$$\Delta_r^* = \begin{cases} 2\sigma_{Ag}^* - \mathcal{W}_r, & \mathcal{W}_r < \sigma_{Ag}^*, \\ \sigma_{Ag}^*, & \mathcal{W}_r \geqslant \sigma_{Ag}^*. \end{cases} \tag{11}$$

Also, the maximum expectation is:

$$E_{max} = \begin{cases} 2\sigma_{Ag}^* - \mathcal{W}_r, & \mathcal{W}_r < \sigma_{Ag}^*, \\ \frac{\sigma_{Ag}^{*2}}{\mathcal{W}_r}, & \mathcal{W}_r \geqslant \sigma_{Ag}^*. \end{cases} \tag{12}$$

We can see that Δ_r^* is dependent on the aggregated variance σ_{Ag}^{*2} and $\mathcal{W}_r$. When the source variation is less, $\mathcal{W}_r$ is small and the compromised node should report $r^{'} = r + 2\sigma_{Ag}^* - \mathcal{W}_r$; in case of a highly varying source, $\mathcal{W}_r$ is large and the compromised node should report $r^{'} = r + \sigma_{Ag}^*$. If there are $\mathcal{K}$ nodes participating in the aggregation process, on an average, a single compromised node is able to distort the aggregated reading r_{Ag} by $E_{max}/\mathcal{K}$, where E_{max} is given by Eq. (12).

Thus, utilizing distribution test and bin test, we are able to restrict significantly the maximum false reading that would be accepted by the system. This is evident from the fact that parameter $2\sigma_{Ag}^*$ in Eq. (11) is typically much smaller in practice than parameter $\mathcal{B}$ in Eq. (3), where $\mathcal{B}$ is the maximum possible difference in the readings that can be observed between two sensors when the phenomenon variation rate is maximum.

C. Compromised Cluster Head

In general, irrespective of the testing scheme being applied, the worst case performance of the system occurs when CH is compromised. This happens because compromised CH can lie about the aggregated report $R_{Ag} = (r_{Ag}, \mu_{Ag}, \sigma_{Ag}^2)$. Here we show the bounds on false readings that can be reported in this case. CH sends R_{Ag} back to selected sensors for endorsement. The following conditions should hold for the R_{Ag} to be accepted for endorsement:

- From Eq. (1), we can see that $\sigma_{Ag} \leqslant min(\sigma_i)$, for each sensor v_i. Hence R_{Ag} with a larger σ_{Ag} will be rejected.

To alter the aggregated mean μ_{Ag}, CH chooses the largest possible σ_{Ag} given by: $\sigma^{'}_{Ag} = min(\sigma_i)$.

- A sensor v_i performs distribution test to test the equality of μ_{Ag} and μ_i. Hence, a large μ_{Ag} will be rejected if it does not satisfy the distribution test. Based on discussion in Section IV-B.1, the maximum false $\mu^{'}_{Ag}$ that can be accepted is given by:

$$\mu^{'}_{Ag} = min\left(\mu_i + z_\alpha \frac{min(\sigma_i^2)+\sigma_i^2}{\sqrt{w}}\right), \qquad (13)$$

  where i is the index of the eligible sensors.
- Further, the aggregated reading r_{Ag} should satisfy the bin test at each endorsing sensor v_i. Let r_{Ag} be the true aggregated reading, and $r^{'}_{Ag}$ be the maximum acceptable false reading reported by compromised CH. It is easy to see that, if CH reports $r^{'}_{Ag} = min(r_i) + 2\sigma_{Ag}$, it will always be accepted. Thus, CH can distort the true readings by a maximum of $r_{Ag} - min(r_i) + 2\sigma_{Ag}$.

On the other hand, with the 1-step Bound Test scheme, when CH is compromised, if CH reports $r^{'}_{Ag} = min(r_i) + \mathcal{B}$, it will always be accepted. Thus, CH can distort the true readings by a maximum of $r_{Ag} - min(r_i) + \mathcal{B}$. As discussed before, since $\mathcal{B}$ is typically much larger than $2\sigma_{Ag}$, our scheme performs much better than simple Bound Test scheme. Also, in our scheme the false $r^{'}_{Ag}$ is always within ±$2\sigma_{Ag}$ of the true r_{Ag}, which is a pretty tight bound.

V. Performance Evaluation

We study the performance of our scheme by simulation. Comparisons are done with the naïve bound test scheme to show significant improvements with respect to immunity to false data injection and with a simple majority voting scheme to show preservation of transient data. We also study the performance of the scheme under various scenarios like different phenomenon variation rates and extent of false injection.

A. Simulation Setup

The wireless sensor network is divided into circular clusters. Each cluster is responsible for sensing the time-varying phenomenon in its region. Cluster nodes are randomly placed in the circular region and one of the nodes is CH. A single source is present at a random location in the cluster. The phenomenon exhibits a radial diffusion pattern, implying that the sensors nearest to the source sense the change first. Table III lists the parameters used for simulation.

TABLE III
Simulation Parameters

Parameter	Notation	Value
Phenomenon variation rate	ρ	10 units/sec
Maximum inter-sensor distance	d	10 meters
Diffusion rate	$\mathcal{D}$	2 units/sec
Sampling rate	x	10 samples/sec
Reporting interval	n	25 samples
Sliding window size	w	1000 samples
Number of nodes in the cluster	τ	10
Random measurement error at sensors		$\mathcal{N}(0, 0.01)$

B. Simulation Results

We conduct various simulations to demonstrate the effectiveness of our scheme and compare it with bound test scheme to show the improvements. We also study the impact of phenomenon variation on our scheme vis-a-vis bin size and the aggregation process.

1) Preservation of Data Transience: We consider the performance of our scheme in the presence of no compromised nodes. The phenomenon varies from 0 units/second to 50 units/second which amounts to a change of 0 units/sample to 5 units/sample. Fig. 9 illustrates the simulation results. In our scheme, most of the times all the nodes form a majority and all the genuine data is preserved regardless of transient variations. It is observed that when the variation rate is small, up to 15% of the nodes are excluded from participating in the aggregation. This could be a possible negative side effect of our scheme, however, since this occurs only when the source data is pretty constant, the effect on r_{Ag} is very small.

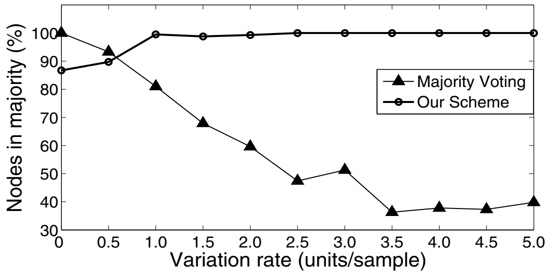


Fig. 9. Nodes in Majority vs. Phenomenon Variation.

We compare our scheme to a simple majority voting scheme where nodes agree if the readings reported are within random measurement error of each other. When there is no variation, the readings are pretty constant and all the nodes agree. However, as variation rate increases, the readings amongst sensors do not agree with each other any more, hence more and more genuine data are excluded from aggregation. Thus the system starts losing "important" information during data transience which is not desired. We can see in Fig. 9, up to 60% data is lost at high variation rates.

2) Immunity to False Data Injection: We consider a single compromised node injecting false data into the network. We study the effect of the false report on the aggregated reading. Also, we simulate how different phenomenon variation rates affect our scheme. Fig. 10 shows the impact on the aggregated reading when when the compromised node is injecting false data at a constant $\Delta = 12.5$ units. The Y-axis represents the effect of false data and is the absolute difference between the true aggregate and the computed aggregate in the presence of a compromised node. It is observed that as the phenomenon variation rate increases, σ_{Ag} also increases (refer to Eq. (12)), and consequently the effect of false data increases.

In Fig. 11(a), we show the effect of false data with respect to different phenomenon variation rates and false injection. The compromised node adds a constant false value, measured in

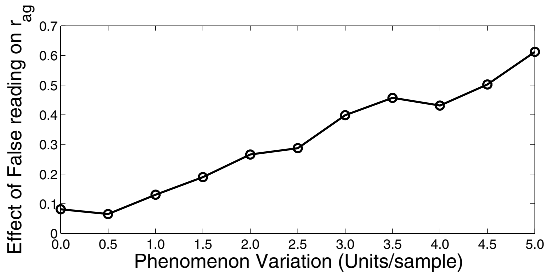
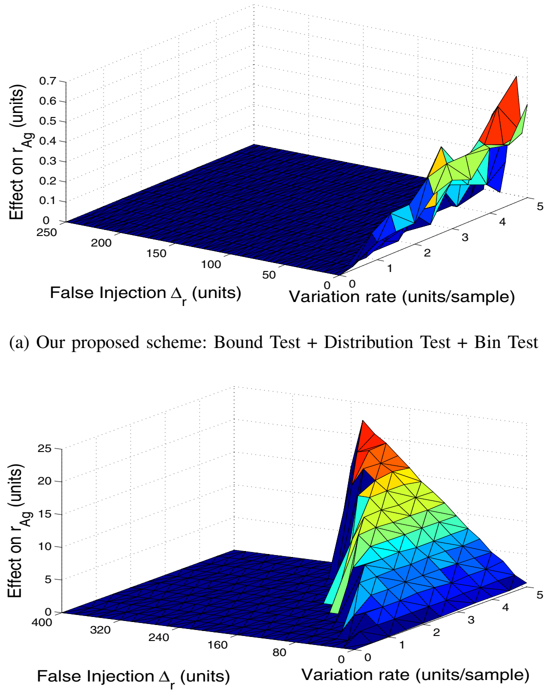


Fig. 10. False effect vs. Phenomenon variation for a constant false injection of $\Delta = 12.5$ units.



(a) Our proposed scheme: Bound Test + Distribution Test + Bin Test

(b) Bound Test scheme

Fig. 11. False effect vs. Phenomenon variation and False Injection. Please note that the z-axis scale of (a) is different than that of (b). The maximum false effect in the case of Bound Test is about 30 times more than that with Bin Test.

units, shown as false injection on the Y-axis. Z-axis represents the effect on r_{Ag}. X-axis represents phenomenon variation rate. As discussed above, it can be seen that, for a constant false injection, the impact of false data increases with phenomenon variation rate. On the other hand for a constant rate, as false injection is increased, the impact of false data first increases and then decreases and becomes zero as the false injection is increased further. This observation is expected and in accordance with our theoretical analysis (refer to Section IV-B.1).

3) Comparison with the Bound Test scheme: We compare the performance of our scheme with the 1-step bound test scheme. Fig. 11(b) shows the effect of false data with respect to varying false injection and phenomenon variation rate. It is observed that the impact of false data increases with higher variation rate since the bound increases. With increasing false injection, the impact of false data first increases and then decreases and finally becomes zero, as expected. However, it is important to note that the effect of the false reading on the aggregate r_{Ag} is much more pronounced in this case than in our scheme. In our scheme, the majority of false injection is filtered out and whatever is accepted has very small effect. In contrast, in the Bound Test scheme the adversary is easily able to inject a difference, approximately 30 times that of our scheme. This indicates the superiority of our scheme and the necessity of using distribution test and bin test.

VI. Conclusions and Future Work

In this paper, we present a robust statistical scheme to monitor transient phenomenon in cluster-based sensor networks. Our scheme requires sensors to add extra statistical information into their data reports to the cluster head. At the cluster head, instead of applying the naive majority voting scheme to test equality of reported sensed readings, three inter-sensor tests, namely bound test, distribution test, and bin test, are performed. As a result, genuine transient data within normal variation range are preserved most of the time, while injected false data are either filtered or their effects on the final data aggregation are limited significantly. We demonstrate the effectiveness of our scheme via detailed analysis and in-depth simulation.

Our scheme presents a complete statistical framework and can be suitably applied on top of any existing security scheme. Future work includes extending our scheme to operate in a dynamic topology, e.g., in conjunction with the security scheme presented in [4]. It would also be interesting to look into the modifications mandated for the application of our scheme to a structure-free aggregation setup [11]. We also plan to implement the proposed scheme on a sensor network test-bed.

References

[1] S. Zhu, S. Setia, S. Jajodia, and P. Ning, “An interleaved hop-by-hop authentication scheme for filtering false data injection in sensor networks,” in *Proc. IEEE Symposium on Security and Privacy*, May 2004.

[2] W. Zhang and G. Cao, “Group rekeying for filtering false data in sensor networks: A predistribution and local collaboration-based approach,” in *Proc. IEEE INFOCOM’05*, Mar. 2005.

[3] Y. Zhang, J. Yang, and H. Vu, “Interleaved authentication for filtering false reports in multipath routing based sensor networks,” in *Proc. IEEE IPDPS’06*, Apr. 2006.

[4] Z. Yu and Y. Guan, “A dynamic en-route scheme for filtering false data injection in wireless sensor networks,” in *Proc. IEEE INFOCOM’06*, Apr. 2006.

[5] F. Ye, H. Luo, S. Lu, and L. Zhang, “Statistical en-route filtering of injected false data in sensor networks,” in *Proc. IEEE INFOCOM’04*, Mar. 2004.

[6] B. Przydatek, D. Song, and A. Perrig, “SIA: Secure information aggregation in sensor networks,” in *ACM SenSys 2003*, Nov. 2003.

[7] L. Hu and D. Evans, “Secure aggregation for wireless networks,” in *Proc. Workshop on Security and Assurance in Adhoc Networks*, Jan. 2003.

[8] A. Mahimkar and T. Rappaport, “SecureDAV: A secure data aggregation and verification protocol for sensor networks,” in *Proc. IEEE GLOBECOM’04*, Nov. 2004.

[9] Y. Yang, X. Wang, S. Zhu, and G. Cao, “SDAP: A secure hop-by-hop data aggregation protocol for sensor networks,” in *Proc. ACM MOBIHOC’06*, May 2006.

[10] V. Shukla and D. Qiao, “Distinguishing data transience from false injection in sensor networks,” to appear in *Proc. IEEE SECON’07*, June 2007.

[11] K. Fan, S. Liu, and P. Sinha, “On the potential of structure-free data aggregation in sensor networks,” in *Proc. IEEE INFOCOM’06*, Apr. 2006.