CHAPTER 5

OUTLIER CLEANING AND SENSOR DATA AGGREGATION USING MODIFIED Z-SCORE TECHNIQUE

5.1 INTRODUCTION

The outlier detection is an important preprocessing routine that is required to ensure robustness of the sensory data analysis. Several factors make the wireless sensor networks (WSNs) especially prone to outliers. A sensor network is equipped with thousands of inexpensive, low fidelity nodes, which can easily generate sensing errors. Since these networks include a large number of sensors, the chance of error accumulates. Besides, in their usage for security and military purposes, sensor nodes are susceptible to manipulation by adversaries. Hence, it is clear that outlier detection should be an inseparable part of any data processing routine that takes place in WSNs.

When there are outliers present in the sensory data, then transmitting these outlier values all the way to the sink is useless, adding additional traffic burden to the already bandwidth constrained radio links. The outlier cleaning process tries to capture the outliers, correct or remove them from the data stream and presents outlier-free data streams to the applications. This chapter examines the use of a simple but efficient statistical test called modified z-score test to precisely label the outliers present in the sensor data during sensor data aggregation process. The efficiency of the proposed aggregation strategy for the two types of traffic models under consideration
viz., (i) continuous data gathering and (ii) event driven, is demonstrated with suitable simulation studies.

5.2 THE AGGREGATION PROCESS IN COLLABORATIVE SENSING APPLICATIONS

Chapter 4 discussed about how the data aggregation done locally at the sensor nodes (regressing the time-series sensor data to construct the AR model) helps in achieving energy efficiency by reducing the communication overhead. In-network data aggregation not only saves energy but it is also warranted for improving the reliability of the sensing results. Since each node has only a limited view of the sensing field and/or the sensing phenomenon, the measurements taken by individual sensor nodes alone are not sufficient and dependable. Therefore, they send the sensed values to an aggregator as shown in the Figure 5.1, which then aggregates the collected reports from individual sensors by applying an appropriate aggregation function depending on the nature of the application and makes a higher-level decision regarding the phenomenon being sensed.

For example, in a typical target detection application, individual sensor nodes collaborate with each other to perform some common tasks like detecting enemy tank movements in defense applications, detecting survivors in disaster rescue operations, tracking animal movements in habitat monitoring applications etc. In such applications, the presence of the target of interest can be considered as the occurrence of the event of interest. Upon detecting the target, the nodes available in the range of the target measure the signal energy emitted from the target and report the readings to the
corresponding CHs as shown in the Figure 5.1. In the aggregation framework shown in Figure 5.1, $x_i$ is the signal energy measured by node $i$ and $Y$ is the aggregated output of the data reported from all the $n$ sensors by applying an appropriate aggregate function $f$. The CH acts as the aggregator and the aggregation function specifies a way to fuse the signals measured at each sensor to produce one consistent and useful result characterizing the whole region. Solutions designed for such collaborative event detection applications are evaluated in terms of percentage of detection and percentage of false alarms (Clouqueur et al 2004a). The detection probability is the probability to decide that a target is present given that a target is in fact present in the region. The false alarm probability is the probability to decide that a target is present given that no target is actually present in the region.

Figure 5.1 The Aggregation Framework Considered for Event/Target Detection Applications
5.2.1 The Need for Robust Aggregation

According to Brooks and Iyengar (1996), when integrating sensor readings, robustness and reliability are crucial properties. A WSN is mostly deployed in potentially adverse or even hostile environments and potential threats may include depletion of batteries, accidental node failures, intentional tampering, failure of communication links and corruption due to noise. Slijepcevic et al (2002) identifies five main sources of errors that influence performance results in the WSNs. Fault tolerance is the ability to sustain sensor network functionalities without any interruption due to sensor node failures.

The inconsistent data reported by the faulty nodes affects the aggregation process and can potentially corrupt the final result, thus requiring the sensor data analysis to be robust against node failures. Hence, dealing with faulty nodes and making reliable decisions in the face of unreliable operating conditions are really challenging and necessitate proper investigation. In this dissertation, an aggregation strategy that uses a statistical test called modified z-score test is suggested for precisely labeling the outliers. This technique uses median of absolute deviation about the median (MAD) in place of the mean and standard deviation that is used in standard z score test.

For the purpose of outlier cleaning, Clouqueur et al (2004) suggested a technique to label the largest and smallest n values reported by sensor nodes as outliers and to drop these values before aggregation; here n is predetermined for a given number of sensor nodes. Such a procedure may inadvertently cause dropping of legitimate values too, thus degrading the detection performance. It is proposed in this thesis that the quality of the aggregation process would definitely improve if a more dependable technique
is adopted for making proper analysis of the distribution of the data to precisely identify the outliers and this inference is verified using simulation. It is seen that the proposed method clearly outperforms the target detection performance reported by Clouqueur et al (2004) and Clouqueur et al (2004 a) as discussed in section 5.6.2.

5.3 DEFINING OUTLIERS

The outliers are the observations that appear to be inconsistent with the remainder of the collected data. The term outlier is used to collectively refer the discordant observations and contaminants. Iglewicz and Hoaglin (1993) defined an outlier as a discordant observation that appears surprising or discrepant to the investigator. Barnett and Lewis (1984) defined a contaminant as an observation from a different distribution than the rest of the data. The possible sources of outliers are also listed in the latter. Recording and measurement errors are often the first suspected source of outliers. In the context of distributed sensor networks used in military surveillance applications, faulty nodes may deliberately introduce some malicious data so as to confuse the aggregation process. An attacker can sometimes generate numerous random or correlated data and inject them into the network. The injected malicious data can drastically change the aggregate data such as average, standard deviation etc; this can also hide the extreme readings when genuine outlier (an extreme deviation from the sample mean) occurs.

5.3.1 The Modified z-score Method

The first step in sensor data analysis is to label the suspected outliers for further study. In a z-score test, the mean and standard deviation of the entire data set are used to obtain the z-score for each data point. An observation is considered to be an outlier, if the sample difference from the
mean is greater than three times the standard deviation (Murray 1998). It should be noted that both the mean and the standard deviation used in standard z-score test are affected by outliers and hence it is inferred that the outlier detection performance will not be dependable. However, in a modified z-score test, the z-score is determined based on outlier resistant estimators. The median of absolute deviation about the median (MAD) is such an estimator which is calculated as follows: (Rousseeuw and Leroy 1987)

$$MAD = \text{median} \{ |x_i - x_m| \}$$

(5.1)

In modified z-score test, the MAD which is calculated using Equation (5.1) is used in the place of standard deviation.

The method includes the following steps:

1. Calculate the sample median \((x_m)\)
2. Calculate the absolute value of the difference between the observations and the median \(|x_i - x_m|\)
3. Calculate the median of the absolute deviation (MAD) about the sample median using Equation 5.1.
4. Calculate the modified z-score for each observation where
   \[ z_i = 0.6745 \frac{x_i - x_m}{\text{MAD}} \]
5. An observation is labeled an outlier when the \(|z_i|\) is greater than 3.5

This is a reliable test since the parameters used to calculate the modified z-score are minimally affected by the outliers. A high modified z-score indicates that an observation is more likely to be a potential discordant outlier. The constant 0.6745 is needed because \(E(\text{MAD}) = 0.6745 \sigma\) for large \(n\).
An often used rule for labeling potential discordant outliers is when $|z_i| > 3.5$ (Iglewicz and Hoaglin 1993).

### 5.4 OUTLIER CLEANING IN EVENT DRIVEN APPLICATIONS

It is assumed that the presence of the target of interest is considered as an event. In this context, the sensor data aggregation is performed on raw energy measurements obtained from individual sensor nodes for making conclusions on the status of a given target. In this work, it is assumed that all the nodes present in the influence field of a given target takes a measurement $x_i$ and reports the observed value to an aggregator. The aggregator’s goal is to apply the modified z-score test to precisely detect the outliers, remove the identified outlier values and then compute an aggregate value $y$ that summarizes the individual nodes’ readings $x_1, \ldots, x_n$, using an appropriate aggregation function $f$. Thus,

$$y = f(x_1, \ldots, x_n).$$  \hspace{1cm} (5.2)

The result of the aggregation process decides about the presence/absence of the target of interest. Clouqueur et al (2004) proposed an aggregation strategy in which the neighboring sensors exchange the measured values among themselves to reach an agreement. In order to alleviate the effect of the outliers on the aggregation performance, the largest and smallest $n$ values of the data exchanged are dropped before the aggregation. The number of values to be dropped is fixed in advance for the given number of sensor nodes as shown in Table 5.1. Cloquer’s method is henceforth referred in this thesis as *drop extremes* method for brevity.
In the proposed work, instead of merely dropping the extreme values, the outliers are more accurately detected using the modified z-score test that well suits to the resource-constrained nature of the WSNs.

Table 5.1  Number of Values to be Dropped (n) for Various Values of N (Cloquer et al 2004)

<table>
<thead>
<tr>
<th>N</th>
<th>9</th>
<th>15</th>
<th>24</th>
<th>36</th>
<th>48</th>
<th>63</th>
<th>81</th>
<th>99</th>
</tr>
</thead>
<tbody>
<tr>
<td>n</td>
<td>3</td>
<td>4</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>11</td>
<td>13</td>
</tr>
</tbody>
</table>

In the above table, N denotes the total number of nodes and n denotes the number of upper and lower extreme values to be dropped.

Another notable constraint in the drop extremes method is its higher communication overhead which is O(k) for each sensor node, where k is the number of its neighbors. This is because their method requires the measured sensor values to be exchanged among the neighbours in order to reach consensus.

Thus, the proposed approach differs from the drop extremes method in,

(i) the way the outliers are detected and handled
(ii) eliminating the need for exchanging values among sensor nodes

Though the proposed aggregation method using modified z-score test involves a slightly higher computational overhead i.e. O(nlogn), where n is the number of sensor data involved in the aggregation process, it offers a
huge savings in communication cost by eliminating the need for data exchange. It may be noted that in the WSNs, the cost of transmitting a bit is many orders of magnitude higher than the cost of executing an instruction. Thus, the slightly higher computational complexity is justifiable considering the huge savings in communication cost.

5.4.1 Aggregation using Modified z-Score Method

The node that performs the role of aggregator (say, a cluster head) performs the following steps in order conclude about the presence/absence of the intended target.

1. Obtain the raw energy values from the sensor nodes.
2. Identify the outliers using the modified z-score test.
3. Remove the identified outliers.
4. Compute the arithmetic mean of the remaining values.
5. Compare the result of step 4 with the chosen threshold for final decision.

Generally, every target will have an influence region within which it is detectable. In the proposed work, the threshold is taken to be the minimum of the signal energy present in the influence field and this threshold is used to measure the detection accuracy and the number of false alarms. The performance of the proposed approach is compared against the drop extremes method and standard z-score method in terms of two parameters namely, percentage of detection accuracy and the percentage of false alarms, for varying number of faulty nodes.
5.5 OUTLIER CLEANING IN DATA GATHERING APPLICATIONS

The previous chapter illustrated how the spatial-temporal similarity existing in the environmental data can be made use of in significantly reducing the communication overhead involved in continuously reporting a huge amount of raw data to the sink. The temporal and spatial similarity has special meaning in outlier cleaning too.

Yongzhen and Lei (2006) differentiated between a simple outlier and a segmented outlier; given a sensing series, a short simple outlier is easy to be identified by human observation because it is shown as a sudden change and extremely different observation from the rest of the data. A long segmented outlier that lasts for a certain time period is not easy to be detected by only examining the sensing series of only one node, because it is hard to tell whether it is a discordant outlier or it represents a change in the data pattern observed. However, according to Yongzhen, the outlier sensor should stand out when compared with the other sensors that monitor the same area because of the spatial similarity of the sensing data. Here, an assumption is made that the sensor nodes are largely and redundantly deployed, and therefore, an environmental change in an area will have similar effect on all the geographically close sensors.

Assuming a feature region consisting of M nodes wherein each node takes a measurement on a feature F (Temperature/light/pressure etc) every T time units during an observation window of size 10. As discussed in section 4.1 of the previous chapter, all the nodes within a given feature region are supposed to communicate similar model parameters to the region head as long as there is no change in the observed phenomenon.
Conversely, one or more nodes may communicate a deviating model to the region head under 3 circumstances:

Case (i) when there is a persistent measurement error (the AR model is updated only when more than 50% of a node’s readings deviate, as mentioned in section 4.4).

Case (ii) the node is manipulated by some adversaries and hence it tries to confuse the system by sending fraudulent and misleading data.

Case (iii) when there is a fundamental shift in the phenomenon being monitored.

Section 5.4 discussed about how the proposed aggregation strategy can be used to detect outliers in target detection applications. For data gathering applications, the aggregation using modified z-score test is done as follows:

Whenever a deviating model is reported by a sensor node, the region head asks the deviating node to send the actual readings and then it forwards them to the cluster head. The cluster head then applies the modified z-score test on the data reported by all the nodes during the observation window, in order to verify whether the deviation is due to a genuine change in the observed phenomenon or is a discordant.

In all the 3 cases mentioned above, the cluster head will be able to accurately identify the observations which are inharmonious with the rest of the data distribution. In cases (i) and (ii), the identified outliers may simply be dropped since they are illegitimate values, whereas case (iii) is to be specially handled by making use of the spatial similarity. The flow of events taking
place at every CH for outlier detection and subsequent re clustering in shown in Figure 5.2.

When more than one nodes report deviating results, the first check to make is to verify whether the defaulters are spatial neighbours (belonging to the same feature region). If so, and if the feature distance of the models reported by those nodes is also within the similarity constant $\delta$, then a fundamental shift in the spatial distribution of the observed feature is confirmed. Subsequently, all these outlier nodes form a new feature region by detaching themselves from their original region. If the spatial similarity is not confirmed, then the reports from those nodes are treated as discordant observations and are dropped by the aggregator.

Figure 5.2 Flow Diagram for Outlier Detection and Re-Clustering
5.6 PERFORMANCE RESULTS - TARGET DETECTION APPLICATIONS

5.6.1 The Simulation Environment

To demonstrate the efficacy of the proposed aggregation procedure, a scenario is simulated where there are N sensors (as per Table 5.1) uniformly distributed over a sensing region of 100 x 100 sq.m area, taking i.i.d. one-dimensional measurements. It is assumed that the signal strength of a target of interest measured at each sensor follows the model (Wei-Peng et al 2004) given below:

\[ R_i = A[D]^{-\alpha} + \omega \]  \hspace{1cm} (5.3)

where,

- \( R_i \) = the received signal strength at the \( i^{th} \) sensor
- \( A \) = signal strength emitted from the target
- \( D \) = estimated distance between the target and the sensor node’s position
- \( \alpha \) = attenuation coefficient; typically ranges from 2 to 5
- \( \omega \) = Gaussian white noise

As shown in Figure 5.3, to measure the detection accuracy, a target is placed at a random position in the sensing field. All the nodes within the influence region of this target now measure their received signal energy level from the target and report them to the aggregator (placed at the centre). Nodes with rings represent faulty nodes. The aggregator now makes a global decision from all the collected data regarding the presence/absence of the target according to the procedure discussed in section 5.4.1.
The faulty nodes present in the network are assumed to follow the Byzantine fault model wherein the faulty nodes send inconsistent and arbitrary data during the aggregation process. According to Lamport et al (1982), Byzantine type of faults encompasses most of the common unintentional and intentional sensor node faults.

The percentage of faulty nodes is varied and the simulations are repeated for varying node densities. It is expected that the proposed aggregation mechanism is able to overcome the effect of the unpredictable and inconsistent values reported by the faulty nodes in deciding the status of the target. The results shown in the graphs in the Figures 5.3 to 5.12 are the averages of values obtained over 50 simulation runs. Subsequently, the target

**Figure 5.3 Sensing Field with Random Distribution of Faulty Nodes**
is removed from the sensing plane and the same procedure is repeated to measure the number of false alarms.

5.6.2 Results and Discussions

After collecting the measured values from all the nodes in the influence field of the target, the aggregator (i) drops the predetermined number of largest and smallest values (as shown in Table 5.1) in *Drop extremes* method (ii) performs the outlier analysis and then drops the detected outliers in z-score and modified z-score methods as discussed in section 5.4.1. Now the performance of the three methods are compared in terms of a) detection accuracy and b) number of false alarms as given below:

As shown in the Figure 5.4, when N=9, it is found that the *drop extremes* method gives slightly better performance than the proposed aggregation strategy, whereas for all other values of N, the proposed method is found to perform much better than the other two as evident from the graphs given in Figures 5.5 to 5.8.
Figure 5.4 Detection Accuracy for N=9

Figure 5.5 Detection Accuracy for N=15
For example, when the total number of nodes is increased to 15 as shown in the Figure 5.5, the drop extremes method guarantees detection up to 33% of faulty nodes, whereas modified z-score method tolerates up to 40% of faulty nodes (with detection accuracy>90%).

Similarly, the performance of the three methods for varying N values is shown in the Figures 5.6 to 5.8. In all the cases shown, it is apparent that as the total number of nodes increases, the performance of the drop extremes method decreases prominently whereas that of the proposed aggregation method increases substantially and remains stable. The drop extremes method does not guarantee a good detection accuracy once the number of faulty values reaches one third of the total node intensity, while the proposed approach shows a greater accuracy (above 90%) up to about 40% of faulty nodes.

Additionally, it shows an average of 37% improvement in detection accuracy against the drop extremes method when the number of faulty nodes is one third of the total number of nodes as shown in the Table 5.2.

### Table 5.2 Average Gain in Detection Accuracy for One-Third of Faulty Nodes

<table>
<thead>
<tr>
<th>N</th>
<th>Drop extremes method</th>
<th>Proposed method</th>
</tr>
</thead>
<tbody>
<tr>
<td>9</td>
<td>100</td>
<td>98</td>
</tr>
<tr>
<td>15</td>
<td>96</td>
<td>100</td>
</tr>
<tr>
<td>36</td>
<td>80</td>
<td>100</td>
</tr>
<tr>
<td>63</td>
<td>20</td>
<td>100</td>
</tr>
<tr>
<td>99</td>
<td>16</td>
<td>100</td>
</tr>
<tr>
<td>Mean</td>
<td>62.4</td>
<td>99.6</td>
</tr>
</tbody>
</table>

Average improvement = 37%
Figure 5.6 Detection Accuracy for N=36

Figure 5.7 Detection Accuracy for N=63
Next the performance of the three methods is discussed in terms of the number of false alarms. As shown in the Figure 5.9, for a very low value of N, the drop extremes method offers a slightly better (12% reduction in false alarms) performance than the proposed approach. However, in all other cases as shown in Figures 5.10 to 5.13, it is found that the proposed approach performs far better than the drop extremes method. Thus, when the number of faulty nodes reaches one third of the total node density, an average of about 48% reduction in false alarms was achieved when using modified z-score method as compared to the drop extremes method as shown in the Table 5.3.
Table 5.3  Average Reduction in Number of False Alarms for One-Third of Faulty Nodes

<table>
<thead>
<tr>
<th>N</th>
<th>Drop extremes method</th>
<th>Proposed method</th>
</tr>
</thead>
<tbody>
<tr>
<td>9</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>15</td>
<td>16</td>
<td>4</td>
</tr>
<tr>
<td>36</td>
<td>52</td>
<td>0</td>
</tr>
<tr>
<td>63</td>
<td>96</td>
<td>0</td>
</tr>
<tr>
<td>99</td>
<td>100</td>
<td>0</td>
</tr>
<tr>
<td>Mean</td>
<td>52.8</td>
<td>3.2</td>
</tr>
</tbody>
</table>

Average reduction = 49.6%

Figure 5.9 False Alarms for N=9
Figure 5.10 False Alarms for N=15

Figure 5.11 False Alarms for N=36
Figure 5.12 False Alarms for N=63

Figure 5.13 False Alarms for N=99
In all the above cases shown in Figures 5.3 to 5.12, the z-score method shows no resistance to even 22% of faulty nodes. This is because, it makes use of mean and standard deviation for outlier detection both of which are affected by outliers. Thus, it is inferred that this method is not reliable for the purpose of outlier detection in target detection applications.

The accuracy of detection is of crucial importance to most of the real world target detection applications. By means of precisely identifying the outliers in sensor data, the proposed aggregation method clearly showcases its potential in improving the aggregation performance of collaborative sensing applications. For easy empirical comparison, the graphs in Figures 5.3-5.7 are presented in the form of Table 5.4 given below.

### Table 5.4 Comparison of Detection Performance of the Three Methods for Varying Number of Faulty Nodes

<table>
<thead>
<tr>
<th>No. of nodes (N)</th>
<th>Detection accuracy (%)</th>
<th>z-score</th>
<th>Modified z-score</th>
<th>Drop extremes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>% of faulty nodes</td>
<td>% of faulty nodes</td>
<td>% of faulty nodes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>33.3</td>
<td>40</td>
<td>45</td>
</tr>
<tr>
<td>9</td>
<td></td>
<td>32</td>
<td>-</td>
<td>24</td>
</tr>
<tr>
<td>15</td>
<td></td>
<td>22</td>
<td>20</td>
<td>16</td>
</tr>
<tr>
<td>36</td>
<td></td>
<td>6</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>63</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>99</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

#### 5.7 PERFORMANCE RESULTS – CONTINUOUS DATA GATHERING APPLICATIONS

With a view to test the effectiveness of the modified z-score test in detecting the outliers in the periodic environmental dataset considered in
chapter 4, an observation window of size 10 is set and this window’s worth of values generated by all the members belonging to a feature region are presented to the proposed algorithm. Feature region 2 in the Figure 4.4 (chapter 4) that consists of nodes 11-15, 17-19 and 21 (9 nodes) is considered for illustration. In this case, for a single observation window, the total data segment is constituted by 90 readings. Different number of outlier values are randomly generated and introduced into the data set and the outlier detection accuracy of the proposed technique is summarized in Table 5.5 given below:

Table 5.5  Outliers Detection in Periodic Data Gathering Network – Statistics

<table>
<thead>
<tr>
<th>Number of Actual outliers</th>
<th>Number of Detected outliers</th>
<th>Percentage of Detected outliers</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>4</td>
<td>100</td>
</tr>
<tr>
<td>9</td>
<td>9</td>
<td>100</td>
</tr>
<tr>
<td>13</td>
<td>12</td>
<td>92.31</td>
</tr>
<tr>
<td>18</td>
<td>18</td>
<td>100</td>
</tr>
<tr>
<td>22</td>
<td>20</td>
<td>90.91</td>
</tr>
<tr>
<td>27</td>
<td>26</td>
<td>96.3</td>
</tr>
<tr>
<td>31</td>
<td>31</td>
<td>100</td>
</tr>
<tr>
<td>36</td>
<td>28</td>
<td>77.78</td>
</tr>
<tr>
<td>40</td>
<td>25</td>
<td>62.5</td>
</tr>
<tr>
<td>45</td>
<td>Nil</td>
<td>0</td>
</tr>
<tr>
<td>49</td>
<td>Nil</td>
<td>0</td>
</tr>
<tr>
<td>54</td>
<td>Nil</td>
<td>0</td>
</tr>
<tr>
<td>58</td>
<td>Nil</td>
<td>0</td>
</tr>
<tr>
<td>63</td>
<td>Nil</td>
<td>0</td>
</tr>
<tr>
<td>67</td>
<td>Nil</td>
<td>0</td>
</tr>
<tr>
<td>72</td>
<td>Nil</td>
<td>0</td>
</tr>
<tr>
<td>76</td>
<td>Nil</td>
<td>0</td>
</tr>
<tr>
<td>81</td>
<td>Nil</td>
<td>0</td>
</tr>
</tbody>
</table>

Total number of data values: 90
From the results of the simulations, it is evident that for up to 35% of the faulty values (31 out of 90), the algorithm is able to accurately label the outlier readings by exhibiting very high detection accuracy (above 90%) as shown in Figure 5.14.

Figure 5.14 Outlier Detection Accuracy for 9 x 10 Readings

Once the outliers are precisely labeled, the concept of spatial correlation is further used to confirm whether they are resulted from reasons (i) and (ii) listed in section 5.5 or due to case (iii) which may then require the formation of new feature region(s) among the nodes observing similar phenomenon. If the cases (i) and (ii) apply, then the labeled outliers may either be replaced by their corresponding values predicted using Equation (4.9) or may simply be dropped so that they do not disrupt the quality of the aggregation process.
5.8 THE GENERALITY AND SCALABILITY OF THE PROPOSED APPROACH

The major factor limiting the performance of any aggregation mechanism is the proportion of faulty values present during the aggregation process. Almost all the sensor network applications fall under either one of the following classes based on the data delivery model employed: (i) Target/Event detection (ii) Query driven and (iii) Continuous data gathering. All these are collaborative in nature and the proposed aggregation strategy is directly applicable for any of the above class of collaborative applications, which is evident from the performance graphs given in sections 5.6 and 5.7.

The simulation results show that the method scales well with increase in node density. Actually, the computational complexity of the aggregation process grows linearly with the size of the network. Yet, in a real setting, this can be managed by enhancing the processing power and storage capacity of the aggregator. Alternatively, multiple aggregators can be employed to share the load of fusing the reported sensor readings.

Figure 5.15 illustrates the aggregation process using multiple aggregators. $s_i$ represents the sensor node $i$, where $i=1,2,...,n$ and the solid circles represents the aggregators. The data from the sensors are routed to the aggregators where it is fused and the fused data is further rerouted. For example, sensor $s_4$ fuses the values $x_1$ and $x_2$ from sensors $s_1$ and $s_2$ to produce the fused value $y_1$; i.e., $y_1 = f(x_1, x_2)$; this process is repeated until the fused data reaches the sink.
Figure 5.15 Aggregation using Multiple Aggregators

Essentially, the node doing the role of aggregator should have some special-purpose hardware that contains the in-built logic to perform the steps involved in modified z-score computation. Also, the aggregator is expected to have higher processing power and storage capacity than individual nodes. Thus the heterogeneous network set up considered in this thesis is well suited for the scenario shown in Figure 5.15.

5.9 SUMMARY

This chapter investigated the possibility of using a simple and effective outlier detection technique using modified z-score method in collaborative sensor networks applications. Mostly, in data processing applications, outliers are discarded from the majority of the data. However, due to the unpredictable nature of the observed phenomena, simply dropping fixed number of values may inadvertently cause losing of important
observations. Therefore, with a view to avoid this risk and aiming towards improving the aggregation performance, a more justifiable and dependable method is proposed in this chapter to precisely identify the outliers in the sensor data. The results obtained from the investigations are promising. Simulations results show that the proposed algorithm yields a very good performance for up to 40% of the faulty node density for target detection applications and for up to 35% of the above in continuous data gathering applications. The proposed method is also found to be scalable as it shows consistent and promising results for large node densities.