**Sproj TITLE**

**Department of Electrical Engineering,**

**Syed Babar Ali School of Science and Engineering,**

**Lahore University of Management Sciences, Lahore, Pakistan**

**Outlier and Event Detection, Identification and Localization in harsh Environments Using Wireless Sensor Networks.**

**Kamran Ali (2013-10-0174)**

**Mohammad Asad Lodhi (2013-10-0175)**

**Ovais bin Usman (2013-10-0026)**

**Syed Bilal (2013-10-0028)**

A report submitted in partial fulfillment of the requirements for the degree of

**BSc Electrical Engineering**

Syed Babar Ali School of Science and Engineering,

Lahore University of Management Sciences

Under the supervision of

**Dr. Ijaz Haider Naqvi**

Designation: Assistant Professor

Email: ijaz.naqvi@lums .edu.pk

Date: 05/2013

Table of Contents

**Acknowledgments**

**Chapter 1: Problem Statement /Introduction**

**Chapter 2: Definitions and Notations**

**Chapter 3: Iterative Algorithm for Outlier Detection**

3.1 The Algorithm

3.2 Optimization of Detection Rate (DR) and False Positive Rates (FPR)

**Chapter 4: Event Detection**

**Chapter 5: Event Identification**

5.1 The Algorithm

5.2 Results and Simulations

**Chapter 6: Iterative Smoothing Techniques**

6.1 The Algorithm

6.2 Results

**Chapter 7: Localization**

7.1 The Algorithm

7.2 Results

**Chapter 8: Conclusion**

**References**

# **Acknowledgments**

First and foremost, we would like to thank to our supervisor of this project, Dr. Ijaz Haider Naqvi for the valuable guidance and advice. He inspired us greatly to work in this project. His willingness to motivate us contributed tremendously to our project. We also would like to thank his for showing us some example that related to the topic of our project. Besides, we would like to thank the authority Lahore University of Management Sciences- EE department for providing us with a good environment and facilities to complete this project. Also, we would like to take this opportunity to thank Sir Aqeel and Nauman Shahid for their constant help to complete this project and see it through. Finally, an honorable mention goes to our families and friends for their understandings and supports to complete this project.

**Chapter 1:**

**INTRODUCTION AND PROBLEM STATMENT:**

Outliers are sensed data measurements in a Wireless Sensor Network that significantly deviate from the normal pattern. [1][2]. The application of outlier detection in filtration of false data, finding faulty nodes and identification of events of interest has attracted significant attention of the research community in recent years. Extensive surveys on characterization and classification of outlier detection techniques [1][3][4] suggest that clustering based algorithms are of significant importance as they are computationally inexpensive, do not require prior knowledge of data distribution, can be used in an incremental model and achieve high detection and low false positive rates. Despite its numerous advantages, the use of clustering based algorithms for Event Detection and Identification has not received extensive treatment in literature. An event can be characterized as an unexpected change in environmental conditions or a hazardous condition for example a fire or gas leakage etc [3]. Event detection and identification, therefore finds application for safety purposes. Formally, a sequence of outliers that show correlation in time and space correspond to an event. Significant research exists on clustering based techniques for outlier detection. For example, the algorithm presented in [5] lays a mathematical foundation for estimating a hyperellipsoidal boundary between normal data and outliers. In [6], the algorithm presented in [5] has been modified to incrementally update the elliptical boundary. The authors of [7] present a way of merging clusters based on their similarity, for global outlier detection. Although these techniques are very effective for outlier detection, they have not yet been used for event detection and identification. Recently, the authors of [8] have proposed an algorithm for event detection by projecting the hyperellipsoids on to single attribute subspaces and analyzing each subspace individually to identify an event, but the algorithm does not accommodate correlation between multiple attributes and is susceptible. Single attribute faults. This paper addresses some of these problems by first presenting an iterative model for hyperellipsoidal clustering and then extending it to a novel Event detection and Identification algorithm. The main contributions of this project can be summarized as follows:

* It presents the mathematical model behind hyperellipsoidal clustering algorithms used for outlier detection.
* It suggests a modification to the clustering algorithm to accurately update the elliptical boundary iteratively.
* It provides a method for optimizing the Detection Rate and False Positive Rate using a Moving Average Filter
* It provides a novel set of conditions that can be used to classify a sequence of outliers as an event.
* It proposes an algorithm for identifying the event by quantifying the relative contributions of the attributes.

**Chapter:2**

**DEFINATIONS AND NOTATIONS**

To describe the hyperellipsoidal model for outlier and event detection we first introduce the required definitions. Let be the first samples of data collected at a node in a WSN where each sample is a vector in. Each element in the vector represents an attribute for example temperature, pressure etc. The sample arithmetic mean or the first moment can be calculated using Eq.1. The second moment is calculated using Eq.2. The covariance in terms of and is given by Eq.3

|  |  |  |
| --- | --- | --- |
|  |  | (1) |
|  |  | (2) |
|  |  | (3) |

The hyperellipsoidal boundary surrounding the data is defined as the set of k data samples whose mahalanobis distance is less than, the effective radius of the hyperellipsoid [7]. can be found by the inverse chi squared distribution i.e. . [1]

|  |  |
| --- | --- |
|  | (4) |

By substituting Eq.3 into Eq.4 we can define the hyperellipsoidal boundary as:

|  |  |
| --- | --- |
| = | (5) |

Eq.5 removes from the equation and represents entirely in terms of means. This step makes it possible to write this equation in an iterative form (Section II).

Outliers are the data samples which are not enclosed in the hyperellipsoidal boundary, Eq.6. Assuming data follows a multivariate normal distribution [2], then up to 98% of the data can be enclosed by choosing an effective radius such that [1].

**Chapter 3**

**3.1 ITERATIVE ALGORITHM FOR OUTLIER DETECTION**

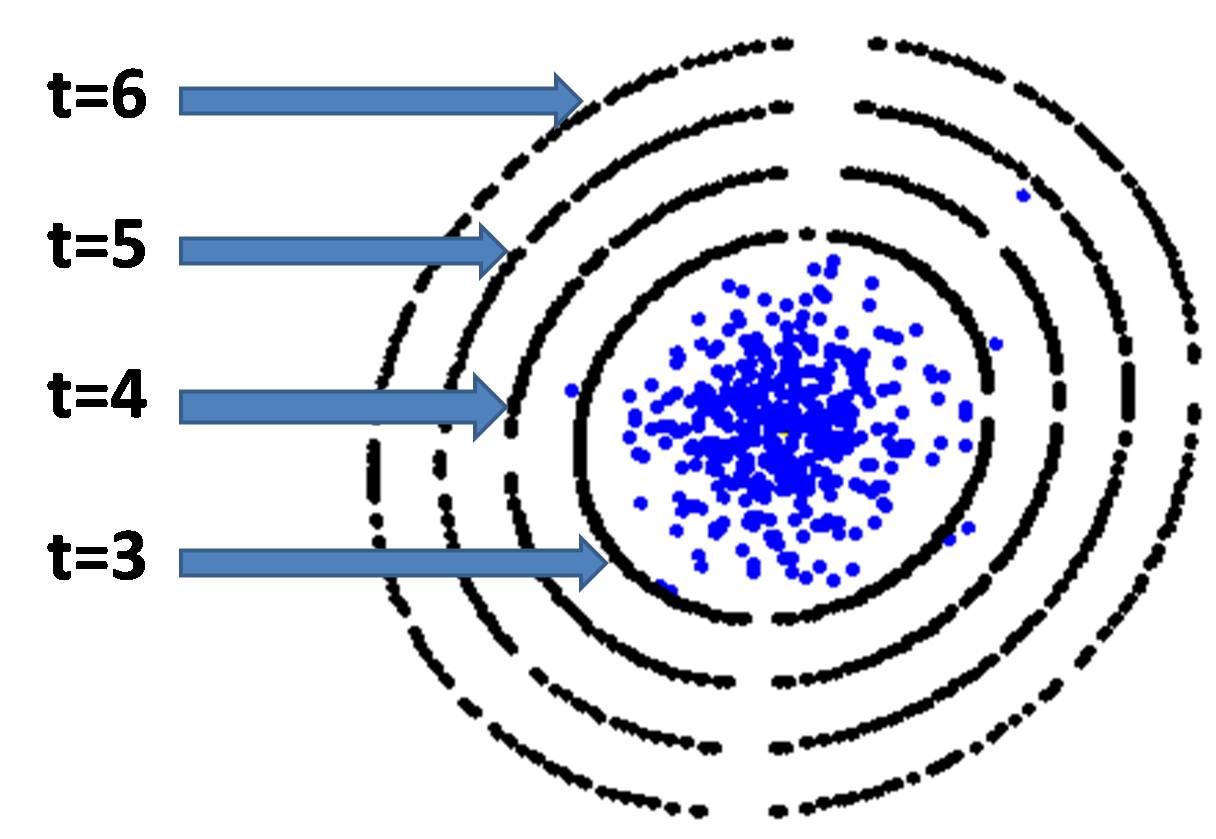
It is often not practical to store all the samples of a streaming data and therefore an iterative algorithm is required to update. Let be the most recent sample recorded at a node. Eq.7 and Eq.8 can be used to update and from the previous means and can be found by using Eq.5. If then it is declared an outlier.

|  |  |  |
| --- | --- | --- |
|  |  | (6) |
|  |  | (7) |

An exponential forgetting factor can be introduced to incorporate the effects of recent measurements Eq.8. [1]

|  |  |  |
| --- | --- | --- |
|  |  | (8) |

It is beneficial to update means instead of covariance because incrementing means requires lower complexity and achieves higher accuracy as compared to the iterative formula for incrementing the covariance matrix. [1]



This Fig shows the effect of changing t on the elliptical boundary



This Fig represents how the elliptical boundary is iteratively updated.

**3.2 OPTIMIZATION OF DETECTION RATE (DR) AND FALSE POSITIVE RATE (FPR) USING A MOVING AVERAGE FILTER**

Random noise in the incoming data is one of the main reasons for degradation of DR and FPR in outlier detection algorithms. The effect of random noise can be mitigated by recursive implementation of a M point moving average filter given by this equation:

where M is the window size of the filter. The value of M should be such that the DR is maximized and FPR is minimized. Hence this results in an optimization problem subject to two constraints. This optimal value for M can be found by modeling the data and simulating the variation in DRs and FPRs with respect to M. The value of that simultaneously optimizes the DR and FPR should then be used to pre-filter the random noise before the data is forwarded for further processing. As shown in the section smoothing the data using a moving average filter remarkably improves the DRs and FPRs for noisy data.

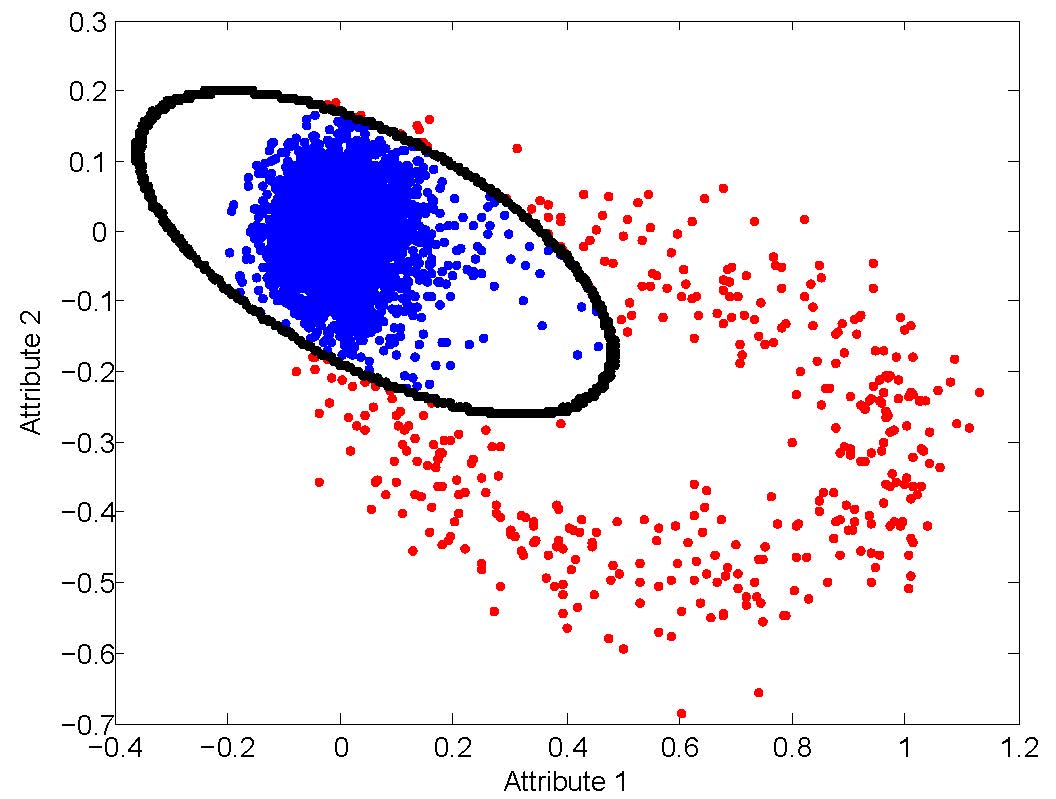
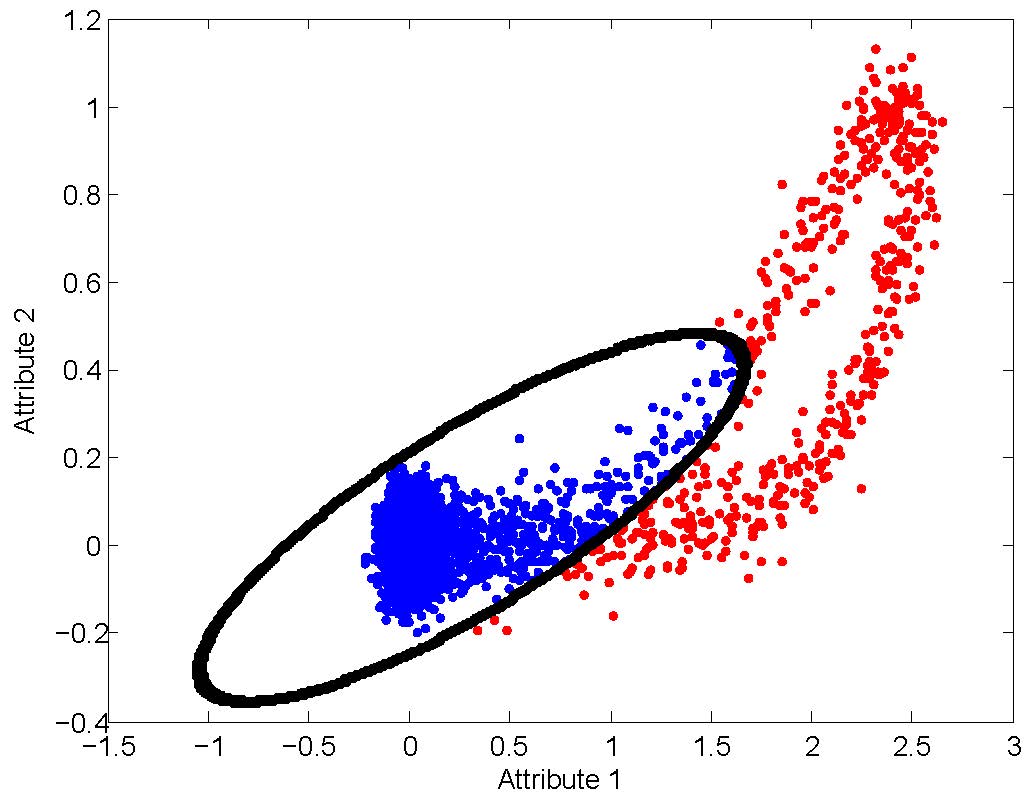
**Chapter 4**

**EVENT DETECTION**

An event is defined as a sequence of outliers correlated in both time and space. To detect an event we therefore need to store the outliers that were detected via the iterative algorithm in section II. Let be the most recent iteration at node, and let be an outlier that occurred at any previous iteration. We define an event array to be an array of outliers that fulfills the following conditions:

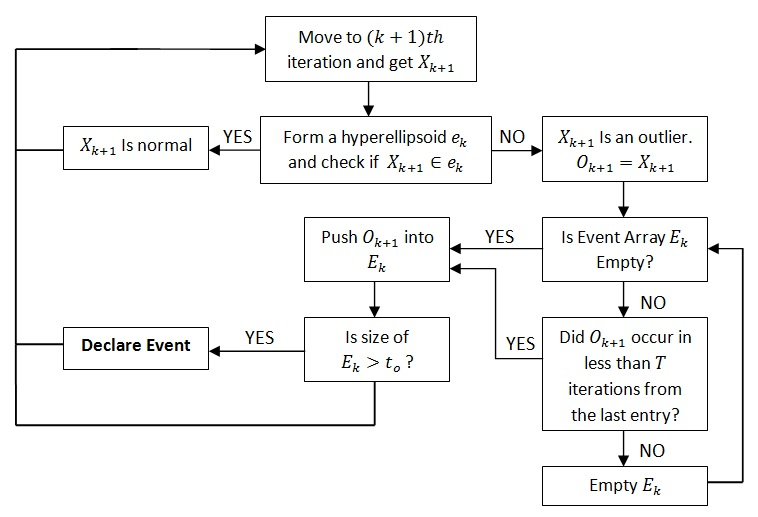
1. A recently detected outlier will be pushed into iff is empty or there is a difference of less than iterations between and the last entry of.
2. If no outlier is detected for iterations, is emptied.
3. If the size of increases beyond a threshold then a local event is declared at node.

These conditions ensure that the outliers based on which an event is declared are not too many iterations apart i.e. they are temporally correlated, and also that they are significant in number i.e. >. The following figure summarizes the algorithm for event detection at a node.

# C:\Users\dell\Desktop\final presentation\Ellipse2.jpg C:\Users\dell\Desktop\final presentation\Ellipse3.jpg

Visualization of hyperellipsoidal clustering



Once a local event is declared, we check if similar events occurred at the surrounding nodes. The similarity can be quantitatively found by clustering at each node and calculating the Bhattacharya similarity coefficient ‘’ between clusters Eq.9 [3][9].

|  |  |
| --- | --- |
|  | (9) |

are means and are the covariance matrices of two clusters and , and is their similarity on a scale of 0 to 1. If between node and its surrounding nodes is high the event at spatially correlated and hence qualifies as a valid Event. Next is forwarded for Event Identification.

Note: Please Move to next page for Chapter 5 Event Identification.

**Chapter 5**

**EVENT IDENTIFICATION**

**5.1 The Algorithm**

In [4], an event identification technique is proposed where is projected on to a subspace which contains only the attribute, and the attribute contributing most towards the event is identified based on deviation of individual projections. In this paper we modify the algorithm by projecting on to a subspace which contains all attributes except the attribute. Hence if is a dimensional column vector, is a dimensional column vector. The advantage of using this algorithm over the single attribute projection algorithm is that attributes are often correlated, and a subspace of multiple attributes ensures that this correlation is incorporated towards the decision. It is also more robust to errors that might occur in measurements of individual attributes. The algorithm is presented in the following three steps:

***STEP 1: Defining the Projection:*** We began by first projecting the arguments of the Mahalanobis distance Eq.6 on.

Let be a matrix with data samples and attributes per sample. We define to be the projection of along subspace. is obtained by removing the row of , resulting in a matrix. gives the projections of along all subspaces, each excluding the attribute.

Similarly, which is a column vector representing the mean of , can be projected on to by removing the row entry. The resulting gives a set of means of each projection in the set.

is a covariance matrix obtained from Eq.3 where the attribute has contribution on both the row and the column**.** is obtained by removing the row and column of the matrix , resulting in a matrix.

***STEP 2: Mahalanobis Distance of the Projections:*** Once projections of all the arguments of the Mahalanobis distance have been found, the Mahalanobis distances can be found simply inserting these arguments into the Eq.10:

|  |  |
| --- | --- |
|  | (10) |

are the projections of Data sample , mean and covariance matrix respectively, on the subspace. For a particular value of, is an array of the mahalanobis distances of each column in for example.

***STEP 3: Decision Formulation:*** The results are based on the deviation of the Mahalanobis distances of the projections from the original Mahalanobis distance. Let be the array of mahalanobis distances of the original dimentional sample. We define to be the decision array for the attribute; it is given by the equation:

|  |  |  |
| --- | --- | --- |
|  |  | (11) |

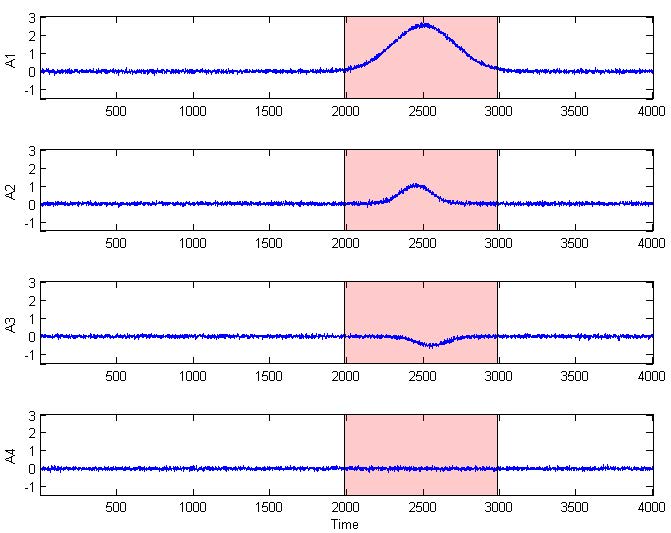
If consists of samples, then the elements of signify the contribution of attribute towards each of the samples. , which is mean of the array gives a measure of the average contribution of towards the event. The attribute that contributes most towards the event is given by:

|  |  |  |
| --- | --- | --- |
|  | Description: C:\Users\dell\Desktop\Research Paper\results\Untitled.png | (12) |

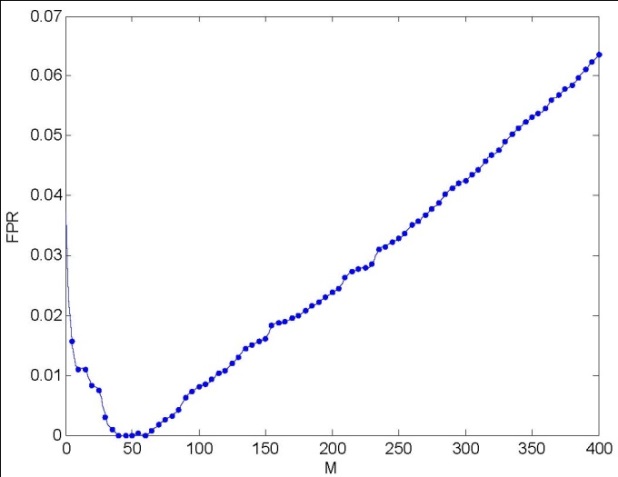
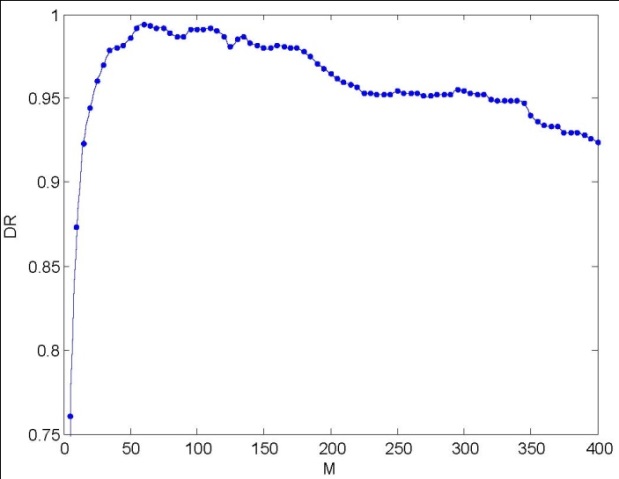
Event Identified using the mentioned algorithm is shown in this figure.

**5.2 RESULTS AND SIMULATIONS**

Simulations were performed in Matlab two a normally distributed 4-dimensional synthetic data sets consisting of 4000 samples each.



Each dimension corresponding to a single attribute is plotted with respect to time in Figure 4 and Figure 5. In the first data set, an event is introduced in the 1st attribute centered at the 2500th sample. The second and third attribute are correlated to the first attribute and hence show slight variations when the event occurs. The 4th attribute is uncorrelated to the other three. In the second data set all attributes are independent and an event occurs in the 1st attribute. A. Finding the optimal value of M M can be optimized by simulating curves for DR and FPR vs. M. the curves for the given dataset are shown in the Figure. It can be seen that both DR and FPR approximately hit their maximum and minimum values at M=50.

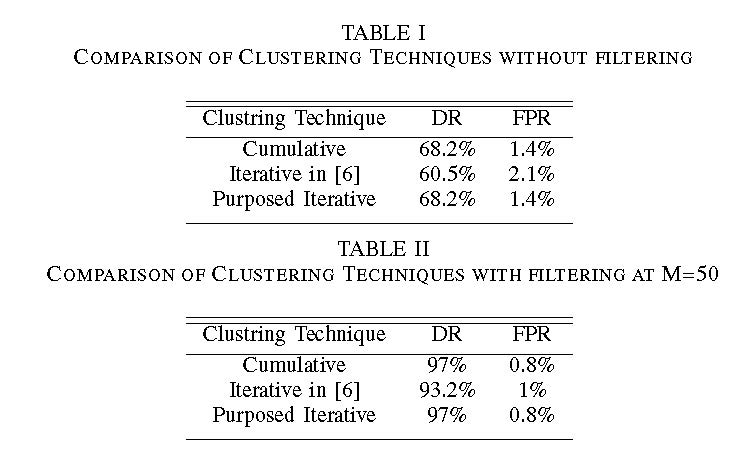


***Outlier Detection:***

Three outlier detection techniques were performed on the Dataset:

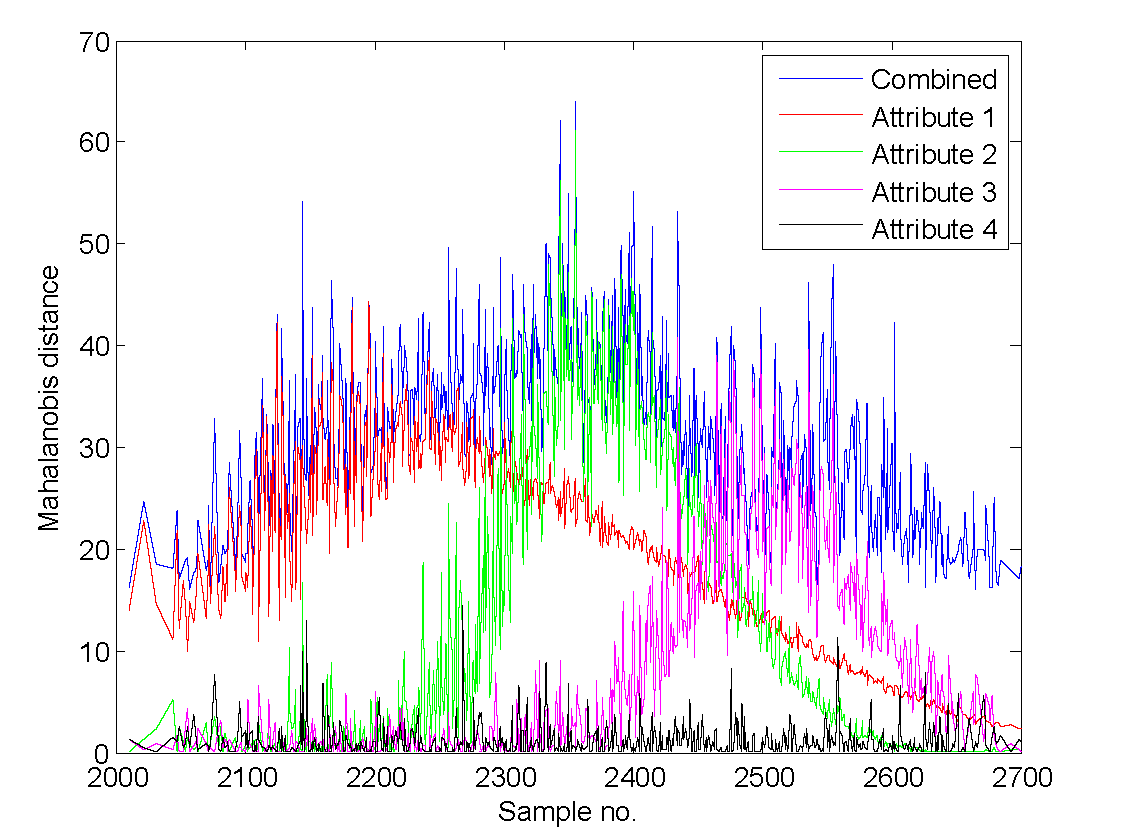
1. Cumulative clustering: To determine if the nth sample is an outlier, all samples until the nth sample are clustered and the Mahalanobis distance for the nth sample is calculated using Eq. (4). This technique is not practical as it would require storage of all n samples which is not possible in case of low memory nodes. However this is the most accurate technique as the knowledge of all the previous data is available while making the decision.
2. Iterative clustering purposed in [6]: The covariance matrix is estimated from the covariance matrix of the previous iteration and substituted into Eq. (4) to make the decision. A loss of precision occurs during the estimation of the covariance matrix and results in a relatively low detection rate (DR) and high false positive rate (FPR) than the other two purposed techniques.
3. Iterative clustering via moments purposed in section II: Here the equation for Mahalanobis distance is simplified in terms of the first and second moment which are incremented with complete precision hence the results for DR and FPR come out to be similar the cumulative clustering technique. As shown in section II, outlier detection only requires the moments of the previous iteration and hence this method is compatible with low memory nodes.

The results for FPR and DR for the three mentioned techniques are summarized in these tables:

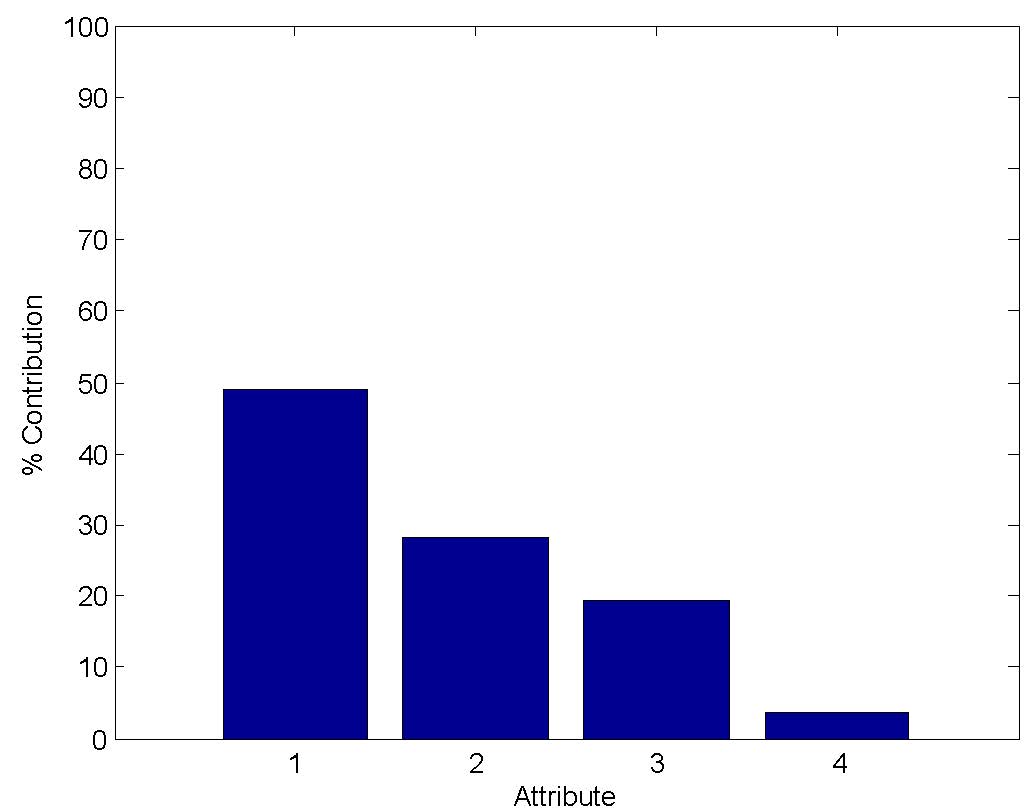


***Event Detection and Identification:***

The Event Detection algorithm mentioned in section IV was applied to the data. Keeping T = 20 and to = 5, an event was recorded with duration from the 2010th sample to the 2700th sample as indicated by the shaded region. This Figure above shows the decision array Rp for p = 1, 2, 3, 4 of the data sets respectively.



The contributions of each attribute over the duration of the event are shown. The event is initiated in attribute 1 which solely contributes towards the event till the 2250th sample after which the effect of attribute 1 dies down and attribute 2 and 3 contribute significantly towards the event. Attribute 4 does not make a significant contribution towards the event throughout the duration of the event. Rkp which is the normalized mean contribution towards the event by each attribute is represented by this Figure.



**Chapter 6**

**ITERATIVE SMOOTHING TECHNIQUES:**

**6.1 The Algorithm**

One of the main reasons for having False Positives in Outlier Detection results is the fast variation in the incoming data. Moreover, some data contain a certain degree of periodicity () and trends. Therefore, there was a need to come up with an algorithm that smoothed the data, thus improving the FPRs and DRs as well.

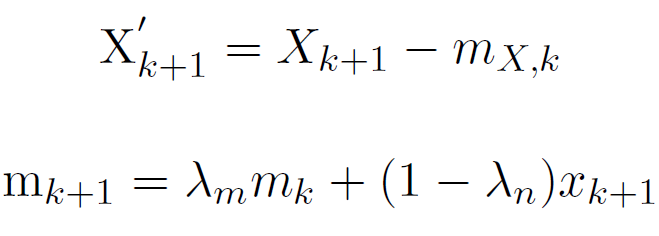
One approach was using the Moving Average (M.A.) Filter was that the optimum size of the window for good FPRs and DRs was data dependent. In case of noisy data like in Figure 1, the results didn’t change much with varying size of the filter. But in data like shown in Figure 2, which include periodicity and trendy variations (an example would be wind blowing in a storm with varying speeds), the FPRs and DRs changed considerably with size of M.A. Filter. Moreover the outlierness of the outliers decreases if larger size windows are used. E.g. if a window size of 25 is used and after 20 normal readings, if 4 or 5 outliers come, the smoothed will value less. The reasons why M.A. filter was not pursued are thus summed up below:

1. The window size is a significant factor which is user defined parameter and gives different results for different readings.
2. The outlier-ness of the outliers is decreased which is bound to affect Detection Rates.
3. Variations like shown in Figure 1 are usually very low compared to the useful data and filters are already implemented in the hardware at DSP level to average/filter such data.
4. Noise as shown in Figure 1 is usually concentrated near mean/center of ellipsoids and due to this reason and the one mentioned in point 3 above such reading are seldom detected as False Positives.

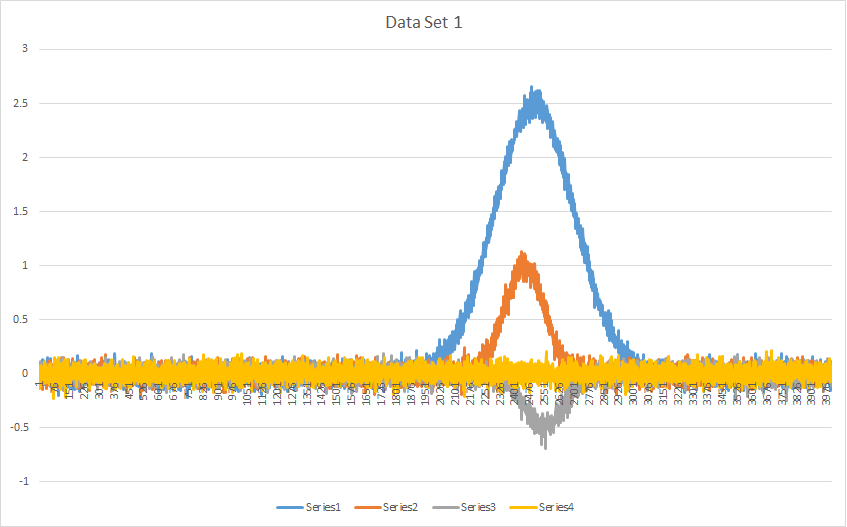
The aim was to bind the data variations as tightly as possible by the hyper-ellipsoid without affecting “outlier-ness” of the outliers. This will lead to minimization of False Positive Rates and increase in DRs.

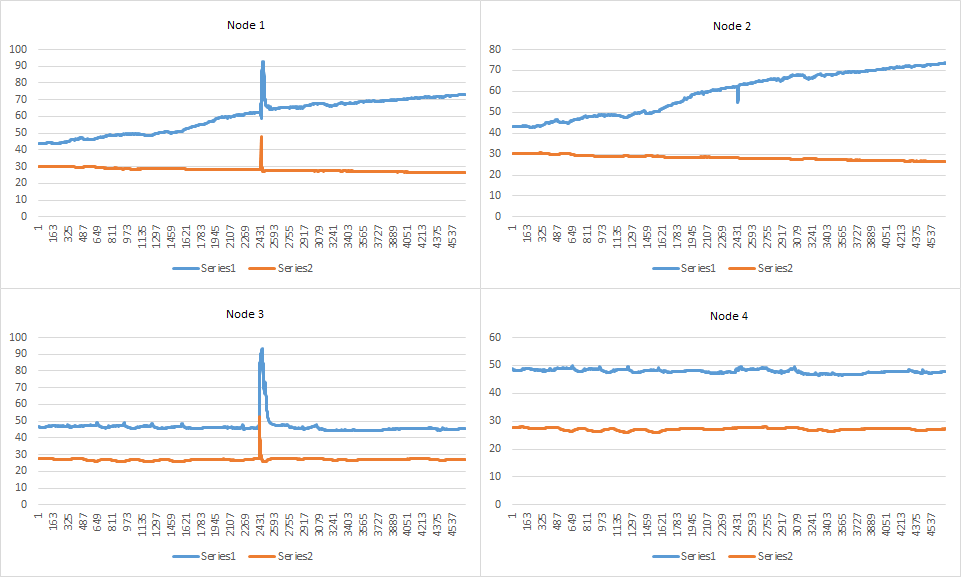
The algorithm developed resulted in excellent results boasting not only low False Positive Rates (FPR) but, at the same time, approaching 100 % Detection Rates (DR). Smoothing and the underlying outlier/event detection algorithm are both still iterative. The technique is named Iterative Smoothing with Independent Forgetting Factors (ISIF).

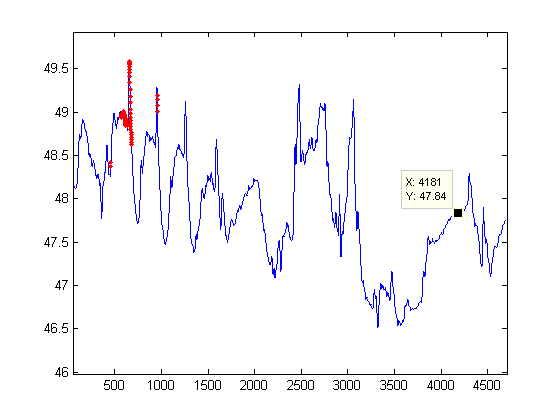
Following is are the equations representing the technique. X’k+1 is the smoothing version of the incoming data, MX,k is the mean updated before X’k+1 and MX,k+1 is the updated mean after the reading. What this method of updating the mean does to the data is shown in Figure 3. Variations become close without affecting much the outlier-ness of a reading. More variations bounded by incremental ellipsoids. This leads to very low FPRs and high DRs.

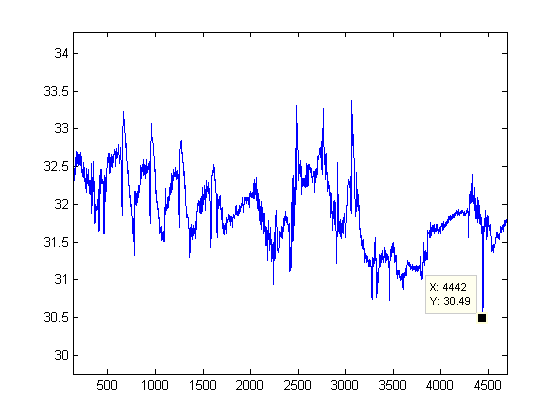


Here λm and λn take values between 0.9 to 0.999.









**6.2 Results:**

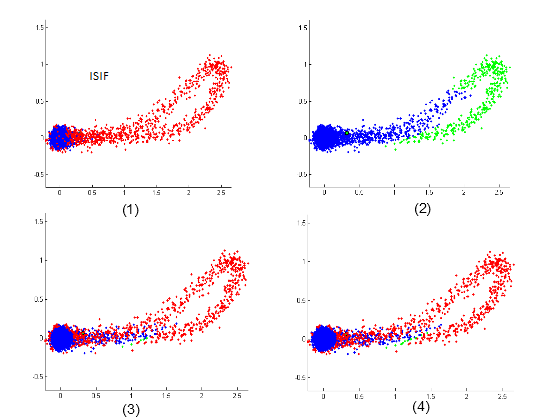
In the start of this section there is an introduction to the data sets used in the evaluation of different methods. Methods with which the results of this algorithm were compared with are as follows,

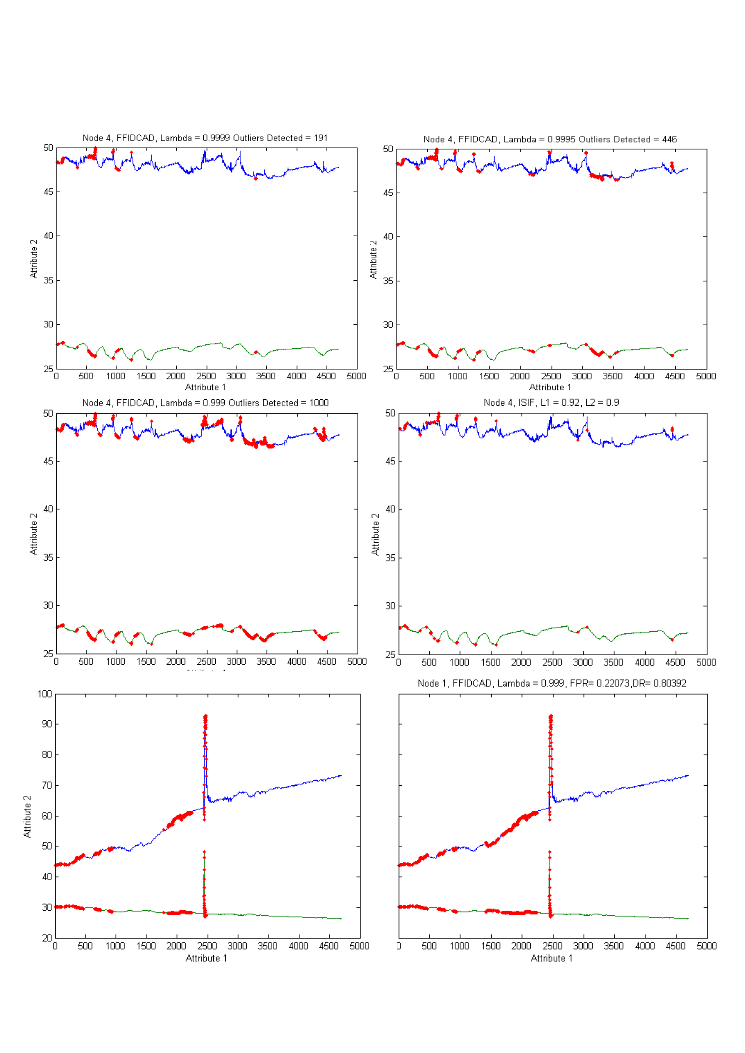
1. Batch Approach - in this approach the Covariance Matrix and the Mean is calculated once for the whole batch or data set.
2. Forgetting Factor (FF) Iterative DCAD (FFIDCAD) Approach - this technique is discussed in the paper by Masudet. al. In this technique both Mean and Covariance Matrix are iteratively updated using a forgetting factor λ. This technique enables the basic IDCAD to track data variation in the monitored environment.
3. N-effective Iterative DCAD Approach - this is another technique presented by Masudet. al. This technique introduces a sliding window based, low complexity approach which is proposed to deal with the effect of a large number of readings on the calculation of

Covariance and Mean.

Figure 4 shows the 2D results of outlier detection on Data Set 1 (attributes 1 and 2 only) using all the four techniques, i.e., the ISIF (1), the batch technique in which the clustering is applied on the whole batch at once (2), the IDCAD without Forgetting Factor (FF) (4) and the

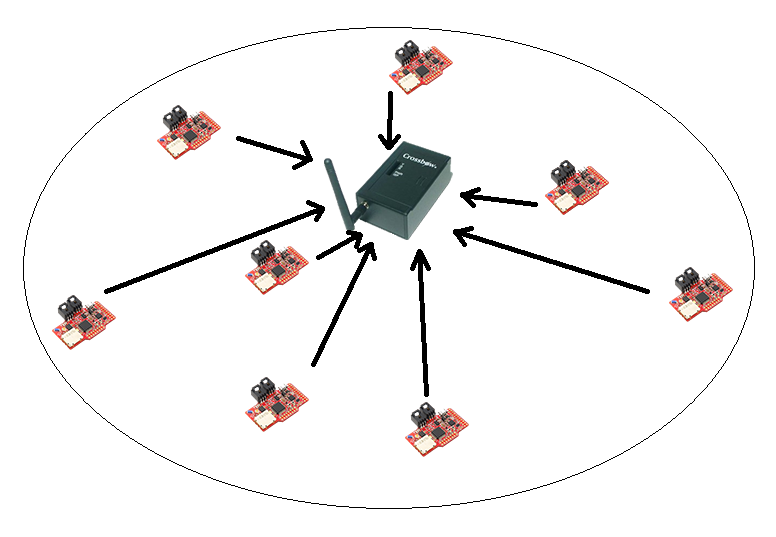
FFIDCAD (3) with FF taken to be 0.999.





**Chapter 7**

**LOCALIZATION:**



**7.1 The Algorithms**

After successful detection and identification of the event our next task was to localize it. For Event Localization we developed a simple technique which uses an iterative approach to localize an event within 10-15 iterations.

***Setup:***

Motes are setup such that there is high power mote at the center of an area of some radius R. This high power mote is capable of doing complex computations and the event localization algorithm will run on this mote. The low power motes in the area sense the event and communicate the event intensities calculated using our event detection algorithm, to the center high power mote. Right now we are not considering the event intensity measured at the center HP mote itself.

Assumptions:

Following are some assumptions behind the algorithm,

a) Intensity of an Event calculated at each node roughly follows a model for event intensity Example

Usual fall-off of event intensity w.r.t distance is

Where *d = distance of mote from the event,*  = fall-off factor (e.g. = 2 for light intensity) and = intensity constant.

b) Event intensity at a certain mote is stable for at-least the sampling interval of the ADC on that mote.

c) Locations of the individual nodes are known at the center mote (e.g. GPS, or any other localization algorithm).

d) The motes are not moving. No mote leaves the environment.

e) Event can be dynamic

Terms:

Here are some of the terms which will be used in the formulation of the algorithm.

* Position co-ordinates

,

* Actual Event L
* Estimated Event L
* No. of nodes = *n*
* is the event intensity communicated to center by each node

***Process:***

* All motes communicate the calculated Event Intensities to the center high power mote
* Center has following information :

1. Positions ,
2. I,
3. The first step in the process is to subtract the mean of the positions from the positions of the motes known at center to get, . This gives new set of positions denoted by an apostrophe sign here. It will become clear in the second step why this was done. The reference becomes the mean of these motes.
4. The second step is that each mote position is given a weight equal to event intensity for that mote and we divide by the sum of calculated intensities. You can now see why we subtracted the mean. This is because if the position co-ordinates of a mote (where intensity of event was low) are of considerably large value w.r.t to the other motes (where intensity of event was relatively high) then that mote can contribute more to this sum in the numerator. Subtracting the mean makes sure that each position’s contribution to the numerator is fair locations of each mote.
5. Iterative Algorithm

For each node information,

1. Using the estimated location of the event, New Intensity is calculated according to the assumed model**. This intensity is the value of event intensity ith mote would have calculated (approximately) if the event had happened at the estimated location.** We have used alpha = 2 and k = 2 in our simulation. So, we calculate by calculating new distance from the estimated event point

\* using *= 2, = 2* for the simulations.

2. Constraint Formulation:

In the 2nd step the constraints are formulated. Using the estimated position and the new distance, we have our first constraint.

+ = --------(1)

The other constraint is the one which depends on the actual event location (which is still unknown) and the actual intensity of the event measured and communicated by ith motes.

+ = -------- (2)

1. Using above constraints we come up with the following constraint which is linear. The square term cancel out. On the RHS you have the known information and on the left you have unknown quantities.

+ = + + = where =

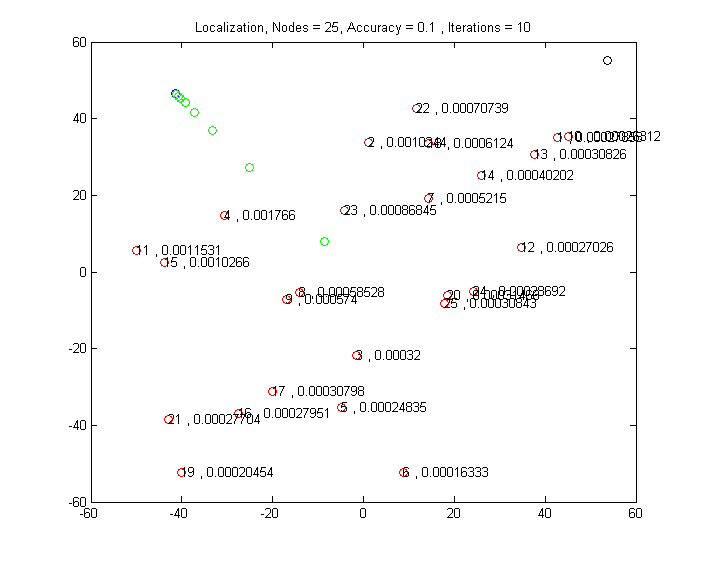
1. For n nodes we get n constraints. This over constrained system can be solved using LSQ method which can be obtained using the well-known pseudo inverse technique. Here P is the position matrix and B is as above. This gives us the new estimate of the event location. We can iterate through these steps until we reach a certain precision, the value of which is set at the start of the algorithm.

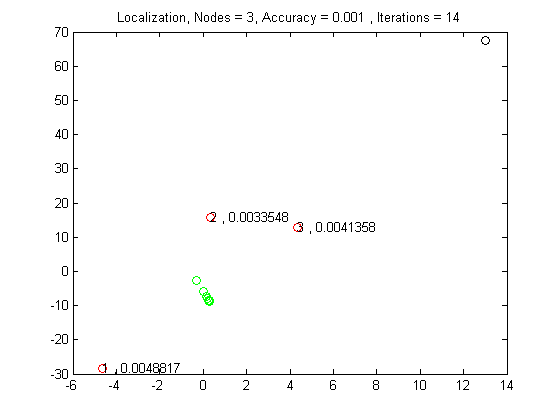
**7.2 Results:**

Here are some of the results for different number of motes and precision values. The observations are,

1. If an event occurs in between a cluster of motes, it needs less iteration to reach the event location.
2. For more precision you need more iterations.
3. The more the number of motes the less iteration it should take to reach the event location.
4. Only 3 nodes are enough to localize an event.

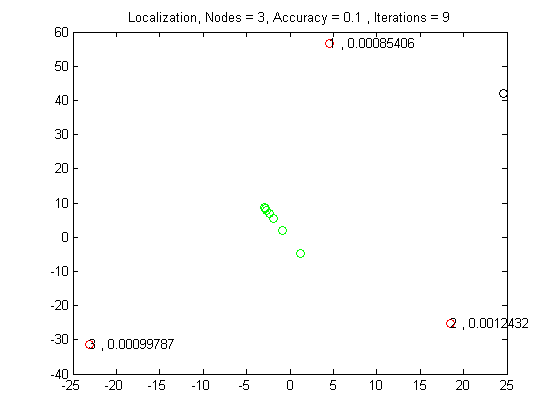
# 





Extension to detect dynamic events:

The algorithm can easily be extended for dynamic events. In that case each mote will be communicating the dynamic event intensities to the center high power mote. The way the algorithm is constructed, we don’t need to make any changes for dynamic event detection. **The algorithm iterative goes through solving the newly formed constraints. Thus it has the capability to automatically track the dynamic event.**



**Chapter 8:**

**CONCLUSION**

In this project the problem of Event detection and Identification via hyper-ellipsoidal clustering has been addressed. First we introduce an iterative method for detection of anomalous data by incrementally updating the first and second moments and using them to find the mahalanobis distance of each sample. The samples whose mahalanobis distance exceeds the effective radius are declared as outliers. The significance of smoothing out the data to remove random noise is also demonstrated. Next we present a novel set of conditions for Event Detection. These conditions check a sequence of outliers for Temporal and Spatial correlation. A sequence of outliers satisfying these conditions is declared as an Event. The contribution of each sensed attribute is then determined by an Event identification algorithm which projects the combined mahalanobis distance on a subspace which neglects the attribute of interest and compares the resulting mahalanobis distance with the combined mahalanobis distance to determine the contribution of that attribute. We also show the effectiveness of the proposed techniques by simulating them on a four dimensional data set and comparing the results with techniques mentioned in the literature. The problems addressed in this project provide the research community with an opportunity for further improve upon the proposed methods and determine

their effectiveness on a variety of simulated and experimental data.

**REFERENCES**

[1] Y. Zhang, N. Meratnia, and P. Havinga, “Outlier detection techniques for wireless sensor . networks: A survey,” Communications Surveys & Tutorials, IEEE, vol. 12, no. 2, pp. 159–170, 2010.

[2] G. H. John, “Robust decision trees: Removing outliers from databases,” in Proceedings of the First International Conference on Knowledge Discovery and Data Mining, pp. 174–179, AAAI Press, Menlo Park, CA, 1995.

[3] N. Shahid, I. Naqvi, and S. Qaisar, “Characteristics and classification of outlier detection techniques for wireless sensor networks in harsh environments: a survey,” Artificial Intelligence Review, pp. 1–36, 2012.

[4] V. Chandola, A. Banerjee, and V. Kumar, “Anomaly detection: A survey,” ACM Computing Surveys (CSUR), vol. 41, no. 3, p. 15, 2009.

[5] S. Rajasegarar, J. Bezdek, C. Leckie, and M. Palaniswami, “Elliptical anomalies in wireless sensor networks,” ACM Transactions on Sensor Networks (TOSN), vol. 6, no. 1, p. 7, 2009.

[6] M. Moshtaghi, C. Leckie, S. Karunasekera, J. Bezdek, S. Rajasegarar, and M. Palaniswami, “Incremental elliptical boundary estimation for anomaly detection in wireless sensor networks,”in Data Mining (ICDM), 2011 IEEE 11th International Conference on, pp. 467–476, IEEE, 2011.

[7] M. Moshtaghi, S. Rajasegarar, C. Leckie, and S. Karunasekera, “Anomaly detection by clustering ellipsoids in wireless sensor networks,” in Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2009 5th International Conference on, pp. 331–336, IEEE, 2009.

[8] N. Shahid, S. Ali, K. Ali, M. Lodhi, O. Usman, and I. Naqvi, “Joint eventdetection & identification: A clustering based approach for wireless sensornetworks,” in 2013 IEEE Wireless Communications and Networking Conference (WCNC), (Shanghai, P.R. China), pp. 1703–1708, apr 2013.

[9] J. Bezdek, T. Havens, J. Keller, C. Leckie, L. Park, M. Palaniswami, and S. Rajasegarar, “Clustering elliptical anomalies in sensor networks,” in Fuzzy Systems (FUZZ), 2010 IEEEInternational Conference on, pp. 1–8, IEEE, 2010.

[10] T. Kailath, “The divergence and bhattacharyya distance measures in signal selection,” Communication Technology, IEEE Transactions on, vol. 15, no. 1, pp. 52–60, 1967.