

# A Fault Management Architecture For Wireless Sensor Network

M.Asim, H.Mokhtar, and M.Merabti  
School of Computing and Mathematical Sciences  
Liverpool John Moores University

M.Asim@2006.ljmu.ac.uk , H.M.Mokhtar@ljmu.ac.uk, M.Merabti@ljmu.ac.uk

**Abstract:** Advancement in wireless communication and electronics has made possible the development of low cost sensor networks. Wireless sensor networks (WSNs) facilitate monitoring and controlling of physical environment from remote location with better accuracy. They can be used for various application areas (e.g. health, military, home). Due to their unique characteristics, they are offering various research issues that are still unsolved. Sensors energy cannot support long haul communication as changing energy supply is not always possible in WSN. Also, failures are inevitable in wireless sensor networks due to inhospitable environment and unattended deployment. Therefore fault management is an essential component of any network management system.

In this paper we propose a new fault management architecture for wireless sensor networks. In our solution the network is partitioned into a virtual grid of cells to support scalability and perform fault detection and recovery locally with minimum energy consumption. Specifically, the grid based architecture permits the implementation of fault detection in a distributed manner and allows the failure report to be forwarded across cells. A cell manager and a gateway node are chosen in each cell to perform management tasks. Cell manager and gateway nodes coordinate with each other to detect faults with minimum energy consumption. We assume a homogeneous network where all nodes are equal in resources. The architecture has been evaluated analytically and compared with different proposed solutions.

## I. INTRODUCTION

Recent developments in wireless communication and electronics have made possible the development of small, inexpensive, low power, distributive devices. These devices are capable of local processing and wireless communication and are known as sensor nodes. A sensor network can be described as a collection of sensor nodes which co-ordinate with each other to perform some specific function. These sensor nodes are mainly in large numbers and are densely deployed either inside the phenomenon or very close to it. A sensor network ensures a wide range of applications. Examples includes environmental monitoring- which involves monitoring air soil and water, condition based maintenance, habitat monitoring, seismic detection, military surveillance, inventory tracking, smart spaces etc. Study of wireless sensor network (WSN) is challenging and requires enormous breadth of knowledge from various disciplines. The design of sensor

network is influenced by many factors, including *fault tolerance* [1-5].

Fault tolerance is the ability to maintain sensor networks functionalities without any interruption due to sensor nodes failure. Fault is an incorrect state of hardware or a program as a consequence of a failure of a component. In order to design efficient fault management architecture for wireless sensor network, various faults associated with wireless sensor network required to be understood. However there is no comprehensive model or description of these faults available [6]. The power supply is the most critical restriction as it is usually difficult to be rechargeable. For this reason faults occurs frequently and will not be isolated events. Attacks by adversaries could happen because these networks will be often embedded in critical applications. Worse, attacks could be facilitated because these networks will be deployed in open spaces or enemy territories, where adversaries cannot only manipulate the environment but gain physical access to the node. Also, communication in sensor networks take place by radio frequencies means that adversaries can easily inject themselves in the network and disrupt infrastructure functions. Moreover, sensor nodes are commonly used to monitor external environment, due to which sensor nodes are susceptible to natural phenomena like rain, fire and fall of trees [6].

Good numbers of fault tolerance solutions are available but they are limited at different levels. Existing approaches are based on hardware faults and consider hardware components malfunctioning only. Some assume that system softwares are already fault tolerant as in [7, 8]. Some are solely focused on fault detection and do not provide any recovery mechanism [15]. Farinaz in [9], grouped faults into 3 types: permanent faults, intermittent faults and transient faults. Permanent faults are the ones that are continues and stable in time. For example, permanent hardware faults are consequences of irreversible physical alteration within a component. An intermittent fault is one that has only occasional manifestation due to unstable characteristics of the hardware, or as consequences of a program being in a particular subset of space. Finally, a transient fault is one that is the consequence of temporary environmental impact on otherwise correct hardware. For example, often the impact of comic radiation may be transient. There is a need of developing a fault management system that

considers different levels of faults in wireless sensor networks. Different mechanisms have been proposed for recovery but still not directly relevant to fault recovery in respect of the network system level management [12].

Given the motivation for applying fault management in wireless sensor networks, we propose a failure detection scheme using grid based architecture. In our scheme, the whole network is divided into a virtual grid where each cell consists of a group of nodes. A cell manager and a gateway node are chosen in each cell to perform management tasks. These cells combine to form various groups and each group promote one of their cell managers to a group manager. Group manager detect faulty cells and avoid future faults. In our scheme both cell manager and gateway node mutually co-ordinate with each other to detect faults inside the cell and perform recovery with minimum energy consumption. We assume a homogenous network where all nodes are equal in resources. Our architecture can be considered as a special kind of clustering architectures. However, it is more systematic, more robust and more scalable.

This paper is organized as follows: in section 2, the general requirements for system design are mentioned. Section 3 provides a brief review of related work in the literature. In section 4, we define the architecture model of our proposed solution and our assumptions for the system. Section 5 describes the management process and types of management messages. Analytic evaluation can be found in section 6. Section 7 concludes the paper.

## II. REQUIREMENTS FOR SYSTEM DESIGN

In this section, we will summarize the most important requirements for a fault management system.

### A. Scalability

Fault management architecture must support scalability of the network. Any increase in network nodes should not affect the overall performance of the network.

### B. Energy Efficiency

Energy consumption is the primary metric and should be considered first. Since sensor nodes are operated on battery and keeping the nodes active all the time will limit the duration that battery last. Therefore, optimal organization and management of wireless sensor network is very crucial in order to maintain sufficient sensors energy to last for the duration of the required task [21]. If a system is not able to maintain sensors energy efficiently then soon it will loose all its energy and fail to operate.

### C. Minimal Data Storage

A sensor node is equipped with only limited memory or storage space. Therefore, the data model used must be extensible and able to accommodate information required to perform management tasks, but also consider the memory constraint of sensor nodes.

### D. Lightweight Operation

Lightweight operation is the fourth important requirement to be considered by a management system. A system should be able to run on sensor nodes without consuming too much memory or energy or interfering with the operation of sensor nodes. Therefore, Lightweight operation prolongs network lifetime [15].

### E. Local detection and Management

Local detection reduces overhead on the network in terms of energy and communication. Also, Local management can be performed through mutual co-ordination of the network nodes. Both local detection and management increases network life time and provide a base for developing an efficient fault management scheme, where management tasks can be distributed over the network and nodes can perform management task locally

## III. RELATED WORK

In this section we will give an overview about existing fault management approaches. Sensor network fault management systems can be classified according to their architectures into centralized and distributed.

In centralized fault management systems, usually a geographical or logical centralized sensor node identifies failed or misbehaving nodes in the whole network. This centralized node can be a base station, a central controller or a manager. This central node usually has unlimited resources and perform wide range of fault management tasks [12]. Examples of these schemes are Sympathy [10], Efficient tracing of failed nodes in sensor networks [11], WinMS [20] and MANNA [6]. Centralized approaches required a central entity to perform the fault management. This creates network overhead as traffic flow is towards a single point. Also, centralized approaches are not scalable and increasing the network nodes affects network performance.

Distributed fault management systems consist of multiple managers. Each manager controls a sub network and may communicate directly with other managers to perform management functions. Distributed management provides better reliability and energy efficiency and has lower communication cost than centralized management systems [15]. Examples include: self monitoring of wireless sensor networks [16], Fault Tolerance Techniques for Wireless Ad Hoc Sensor Networks [9], Localized fault-tolerant event boundary detection in sensor networks [22] and Clustering [17]. These solutions are based on neighbouring co-ordination, self monitoring or clustering. In neighbouring co-ordination, nodes coordinate with their neighboring nodes to detect faulty nodes before contacting the central point. For Example, the algorithm proposed for faulty sensor identification in [22] is based on neighbouring co-ordination. In this scheme, the reading of a sensor is compared with its neighboring' median reading, if the resulting difference is large or large but negative then the sensor is very likely to be faulty.

Chihfan et. al [16] developed a Self monitoring fault detection model on the bases of accuracy. This scheme does not support network dynamics and required to be pre configured.

Clustering is an efficient approach for building scalable and energy-balanced applications. The failure detection solution proposed in [17] is based on cluster formation. It divided the network into different clusters and a cluster head is appointed for each cluster. A cluster head has more resources than other cluster members. This cluster head perform major tasks and management operations. The data can only be transmitted to other clusters through cluster heads. Failure of a cluster head limits accessibility to the nodes under its supervision. In most of clustering approaches, cluster head ask its cluster member to send there updates on regular basis. Upon receive a request from cluster head, cluster members send there updates to there respective cluster heads. If a cluster head does not receive a reply from any node then it considers that node faulty. This causes vulnerability to message loss because a node can not receive a request message from its cluster head due to various reasons. It can be lost during transmission and cause a correct node to be declared as faulty. This violates the rule of accuracy.

#### IV. A NEW CELLULAR FAULT MANAGEMENT ARCHITECTURE

In this section we present our new architecture for fault management in WSN: the cellular architecture. In this architecture we divide the network into a virtual grid of cells. A cell can be considered as a special kind of clustering. However it is more systematic and scalable. Cells can merge together to produce large cells that would be managed using the same process.

As discussed above that fault can be permanent, intermittent or transient and in our fault model we only consider permanent faults only. We assume that communicated data is fault free and that all *semantic-related generic faults* are detected and removed by the application itself. Furthermore, we assume that there will be no alterations or creations of messages over the transmission links.

Our aim is to distribute the management task across the whole network so that management tasks can not be performed by any central entity or a part of the network. Therefore, Sensor nodes will take more management responsibilities and decision making in order to achieve a self managed network. Division of network into virtual grid helps in achieving self configuration, in which it must actively measure network states in order to react to the network dynamics.

We uses the same architecture proposed in [19] but for the purpose of fault management in wireless sensor networks. In our proposed solution the whole network is divided into a virtual grid using some type of virtual coordinate system.

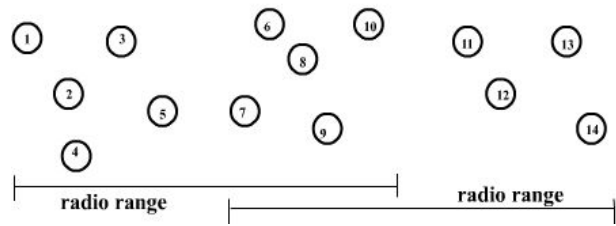


Fig. 1. Sensor nodes

The grid architecture can be achieved by dividing the whole area where nodes are distributed into small virtual cells. In Fig. 1, nodes 1 to 10 are in the communication range of each other so it can be divided into two adjacent virtual cells. The division will be made in such a way that all nodes in Cell A can communicate with all nodes in adjacent Cell B.

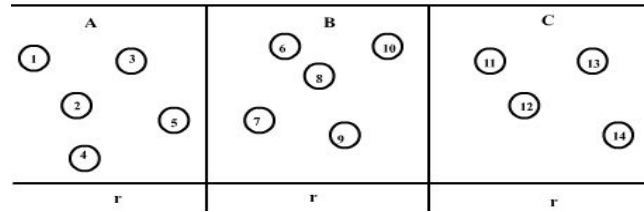


Fig. 2. Division of the network into a virtual grid

After the division of the network into small virtual cells as shown in Fig. 2, a cell manager is appointed in each cell. The cell manager then selects a gateway node and decides which common node will perform sensing and which will go to sleep. This would help in conserving energy. The cell manager receives data from common nodes and passes it to the gateway node which in turn passes it to other gateway nodes.

##### A. Distributed Cell Formation

In order to elect cell managers and gateway nodes, any node can send a discovery message that consist of its node ID, Cell ID, and energy level and only nodes with higher energy would respond. The combination of Cell manager with gateway node guarantees a connected network.

##### 1) Cell manager selection

The node with the highest life time or energy will be appointed as a cell manager. Cell manager keep changing in each cell in order to extend network life time.

##### 2) Gateway node selection

Cell manager will select the gateway node on the basis of Maximum energy. Communication between cells takes place through gateway nodes. The cell manager broadcast a message to its cell members which in return send their updates including their node ID and energy level. Upon receiving updates from cell members, the cell manager appoints a gateway node. Both cell manager and gateway node stores node ID's of their cell members.

### 3) Group manager

After the selection of cell managers and gateway nodes, cells combine to form various virtual groups. Each group of cells then selects a group manager with mutual co ordination. A group manager is a cell manager which performs its normal tasks for its own cell but at the same time act as a group manager for a group of cells. This is shown in figure 3.

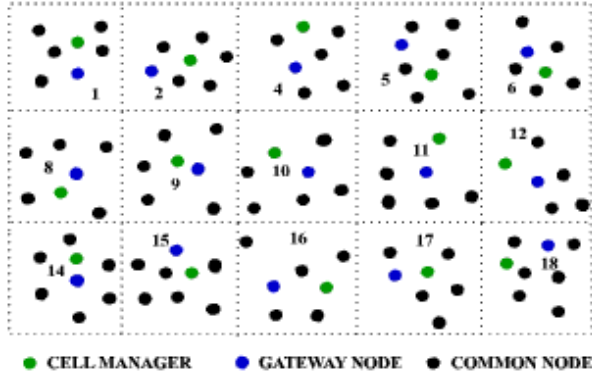


Fig. 3. Virtual cells in the form of a grid

Cells 1,2,8,9 combine to form a group and with mutual co ordination they promote cell 9 cell manager to a group manager. Cell 9 manager will now perform two roles; one as a cell manager to perform management tasks for its own cell and second as a group manager for a group of cells. The main goal of introducing this group manager is to perform high level management tasks and predict future faults. The selection of group manager is based on the available energy and it keeps changing in the group to balance the energy. Each cell maintains its health status in terms of energy. It can be High, Medium or Low. These health statuses are then sent out to there associate group managers. Upon receiving these health statuses, group manager predict and avoid future faults. For example; if cell 1 has health status high than group manager always recommend that cell for any operation or routing but if the health status is medium than group manager will occasionally recommend it for any operation. Health status Low means that the cell has un-sufficient energy and should be avoiding for any operation. Therefore, a group manager can easily avoid using cells with low health status. Group managers keep updating each other regarding their group member's health statuses.

## V. MANAGEMENT PROCESS

Our proposed fault model covers different types of faults. One of the most common faults is caused due to power supply as it is not rechargeable. Once a node power supply is expired then it can not participate in network operations. This type of fault can be rectifying by sending the node in low computation mode before it completely shut down.

Our proposed solution constantly monitors sensor nodes energy level and makes them sleep if they are low in energy. Also, sensor nodes are commonly used to monitor external

environment and are susceptible to natural factors like fire, rain and fall of tree. This type of effect can permanently damage the sensor node. Our proposed model can identify these faulty nodes and replace them with back up nodes. Another common type of fault can be cause by adversaries as sensor nodes will often be deployed in open space or enemy territories. Our proposed solution can detect future faults through group managers, which identify the nodes with low energy and inform the network to avoid using them.

Adversaries can easily disrupt communication by jamming or manipulating the network. In our proposed solution node can co-ordinate to identify these types of faults by invoking security procedures. Fault can also be caused by declaring a correct node faulty, as this affect network operation.

### A. Fault detection

The fault detection process can be divided into:

In-Cell fault detection

Cell and Cell manager's fault detection & recovery.

#### 1) In Cell fault detection

Cell manager ask common nodes and gateway nodes on regular bases to send there updates. Such as; the cell manager send get messages to the associated common nodes and gateway node on regular basis and in return they send there updates. Updates include node ID and energy level. Gateway node also sends updates to cell manager like any other common node. If the cell manager does not receive an update from any node then it send an instant message to the node and acquire about its status. If cell manager does not receive the acknowledgement in a given time then declare it as a faulty node and pass this information to the rest of the network. Using updates, cell manager decide which node will go to sleep and which will remain active. If the gateway node is low in energy then cell manager appoint another node to act as a gate way node and send the existence gateway node to low computational mode or sleeping mode. The gateway node can also be detected as faulty, if the cell manager does not hear from it during updates cycle. The cell manager appoints a new gateway node and informs the neighbouring cells and the group manager.

#### 2) Cell and Cell manager's fault detection & recovery

Each cell manager sends health status information to its group manager through gateway node. This is less frequent than in-cell update cycle. If the gateway node does not receive the get message from its cell manager during updates cycle then it send a quick reminder to its cell manager and expect a quick reply but if it does not receive any reply then it consult common nodes and inform them about the failure of the cell manager and ask them to send there current status information. Upon receiving updates from common nodes, it appoints a new cell manager and informs its group members.

If group manager doesn't receive health status from a particular cell then it waits for the second health status update. If a group manager doesn't receive the health status updates during the second cycle then it informs the whole network about the occurrence of a faulty cell. A group manager can be

replaced by another cell manager in case of a fault or energy reason.

### *B. Management role*

In centralized management system, the base station acts as a central manager and control the entire network. It collects information from all nodes and performs complex management tasks. The central manager has a global view of the network and with unlimited resources. Distributed management system employs multiple manager stations. Each manager is then responsible for a sub network and able to communicate directly with other manager to perform management task. A hybrid between centralised and distributed is hierarchical management system. It uses intermediate managers which do not communicate with each other directly. Each manager controls a sub network and passes information from its sub-network to its higher-level manager. Intermediate managers also disseminate management functions received from higher-level managers to their sub-network [15].

In our proposed solution, both cell manager and gateway node perform management task with mutual co-ordination. They are responsible for the members of their own cell. The main emphasize is to perform fault management locally. Both cell manager and gateway node are at the same management level. Cell manager detect faults inside the cell and gateway node takes part in routing information and monitoring its cell manager. Both cell manager and gateway nodes can be replaced by common nodes in case of backup and recovery. Another layer of management comprises of group managers, which are responsible for detecting faulty cells and predict future faults.

### *C. Types of Management Messages*

Each Cell manager send get messages to their cell members including gateway nodes on regular basis. Upon receiving the get message, cell members reply with their updates including node ID and energy status. This is called the in-cycle updates. Another type of message exchange takes place between the group manager and its group members. This is less frequent than in-update cycle and called health-status updates. This is used to monitor the functionality of overall cell. If a group manager doesn't receive a health status update from a particular cell then it consider it a suspicious cell and wait for another health-status update from the same cell. If it doesn't receive the second health status update from the same cell then it announce it a faulty cell.

## VI. ANALYTICAL EVALUATION

In this section our proposed architecture has been evaluated analytically and compared with some existing proposed solutions.

Our proposed solution can be considered as a special type of clustering but more systematic, robust and scalable. Clustering has been used to address various issues i.e. routing, energy efficiency, management and huge-scale control. Therefore clustering can be formed in several ways. Nodes generally

form a cluster in two stages: (1) a header is selected among the nodes through election algorithm, randomized election, degree of connectivity or pre-definition, and (2) the headers and the nodes interact to form a group or a cluster [23]. Cluster heads are responsible for coordinating the nodes in their clusters and generally are more resourceful than its cluster members. Cluster heads are the traffic bottlenecks; there failure may cause several problems. Also, if a cluster head failed to operate then no messages of its cluster will be forwarded to the base station and selection of the new cluster head is energy consuming.

Our proposed architecture also divides the network into small virtual cells and each cell consists of a group of nodes. Unlike clustering, it is homogenous network where all network nodes are equal in resources. We also appoint a cell manager for each cell to carry out management tasks with mutual co ordination of gate way node but a cell manager can easily be replace by any other common node if the cell manager stop operating. Gateway node is responsible for routing information to other cells and if a cell manager get faulty than gateway node start acting as both cell manager and gateway node until it appoint a new cell manager. Also, if a gateway node stops operating than the cell manager will take its place. Therefore, network operate normally even in the presence of a fault.

Another important factor that needs to be considered is vulnerability to message loss. In clustering, if a cluster head does not hear from its cluster member than it announced it as a faulty node. However, a message can be lost due to various reasons. It can be lost during transmission and cause a correct node to be declared as faulty. In our proposed architecture, a cell manager also ask its cell members to send there updates regularly. If a cell manager does not hear from a node then it send another quick message to the node and wait for a reply. But if the cell manager does not receive the reply message for the second time then it declared it as a fault. We check the suspicious node twice so that a correct node can not be declared as faulty.

.Our proposed solution is a distributed approach and easily scales with the growth of the network. Unlike Centralized solutions, there is no central manager and all management tasks are performing locally and in distributed fashion. It is not based on any historical data or global view of the network to detect faults.

Energy-efficiency is an important research challenge to succeed the vision of a self organized wireless sensor network. Our approach addresses this challenge by employing a load balancing strategy so that all nodes operate together for as long as possible. We consider that all the nodes in the network are equal in resources and no node should be more resourceful than any other node. The optimal role assignment and reconfiguration scheme support the network management system to utilize the network nodes in the most efficient manner. Our approach does not rely on specific nodes with extra resources but assign tasks due to there optimal capabilities. Nodes are ranked according to their available energy. Therefore, the selection of the gateway node and a cell

manager is based on the available energy. The basis idea of this design is to encourage nodes to be more self-manageable and extend the network life time for as long as possible. Also, distributed management system has lower communication costs and provides better reliability and energy efficiency.

It is a simple grid base architecture which supports lightweight operations. The main idea to propose a model that doesn't consume much energy or memory. It is based upon the virtual coordination due to which cells are built automatically and there is no need to exchange too much messages to build the clusters. The cell member remains on the same cell regardless of the cell manager.

Our architecture divides the whole network into a virtual grid and enables the network to perform local detection and distribute the management tasks across the network. This approach helps sensor nodes to take more management responsibility and decision-making in order to success the vision of self managed WSNs. Also, this increases network life time. The cellular architecture is for management purpose only so they can be merged into clusters for routing or any other purpose if needed.

## VII. CONCLUSION

In this paper a new cellular fault management scheme is presented to achieve fault management efficiently. We divided the network into a virtual grid, where each cell consists of a group of nodes. This supports scalability of the network and increase network life time. Most of existing solution used some type of central entity to perform fault management tasks but in our proposed solution, the aim is to perform fault detection locally and in distributed fashion. Unlike clustering, it is based on homogenous paradigm where nodes are of equal resources and can easily back up each other in case of recovery.

## REFERENCES

- [1] A. Bharathidasas, and V. Anand, "*Sensor networks: An overview*", Technical report, Dept. of Computer Science, University of California at Davis, 2002
- [2] D. Estrin, R. Govindan, J. Heidemann, and S. Kumar, "*Next century challenges: Scalable coordination in sensor networks*", in Proceedings of ACM Mobicom, Seattle, Washington, USA, August 1999, pp. 263--270, ACM.
- [3] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci, "*A Survey on Sensor Networks*", IEEE Communications Magazine, pp. 102--114, August 2002.
- [4] D. Estrin, L. Girod, G. Pottie, M. Srivastava, "*Instrumenting the world with wireless sensor networks*", In Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP 2001).
- [5] E. S. Biagioni and G. Sasaki, "*Wireless sensor placement for reliable and efficient data collection*", in the 36<sup>th</sup> International Conference on Systems Sciences, Hawaii, January 2003.
- [6] L. B. Ruiz, I. G. Siqueira, L. B. Oliveira, H. C. Wong, J.M. S. Nogueira, and A. A. F. Loureiro. "*Fault management in event-driven wireless sensor networks*", MSWiM'04, October 4-6, 2004, Venezia, Italy
- [7] J. Chen, S. Kher and A. Somani, "*Distributed Fault Detection of Wireless Sensor Networks*", in *DIWANS'06*. 2006. Los Angeles, USA: ACM Pres.
- [8] F. Koushanfar, M. Potkonjak, A. Sangiovanni-Vincentelli, "*Fault Tolerance in Wireless Ad-hoc Sensor Networks*", Proceedings of IEEE Sensors 2002, June, 2002
- [9] F. Koushanfar, M. Potkonjak, A. Sangiovanni-Vincentelli. "*Fault tolerance techniques in wireless ad-hoc sensor networks*", UC Berkeley technical reports, 2002.
- [10] N. Ramanathan, K. Chang, E. Kohler, and D. Estrin, "*Sympathy for the Sensor Network Debugger*", in 3<sup>rd</sup> Embedded networked sensor systems. 2005. San Diego, USA: ACM Press.
- [11] J. Staddon, D. Balfanz and G. Durfee, "*Efficient Tracing of Failed Nodes in Sensor Networks*", in *First ACM International Workshop on Wireless Sensor Networks and Applications*. 2002. Atlanta, GA, USA: ACM.
- [12] M. Yu, H. Mokhtar, M. Merabti, "*A survey on Fault Management in wireless sensor network*", in: Proceedings of the 8th Annual PostGraduate Symposium on The Convergence of Telecommunications, Networking and Broadcasting, 2007.

- [13] F. Koushanfar, M. Potkonjak, and A. Sangiovanni-Vincentelli. "On-line Fault Detection of sensor Measurements", Sensors, 2003. Proceedings of IEEE Volume 2, 22-24, Oct. 2003, pp.974-979
- [14] J. Chen, S. Kher, and A. K. Somani, "Distributed Fault Detection of Wireless Sensor Networks," in Proc. of DIWANS 06, Sept.2006, LA
- [15] W. L. Lee, A. Datta, and R. Cardell-Oliver, "Network Management in Wireless Sensor Networks", to appear in Handbook on Mobile Ad Hoc and Pervasive Communications, edited by M. K. Denko and L. T. Yang, American Scientific Publishers.
- [16] C. Hsin and M.Liu, "Self-monitoring of Wireless Sensor Networks", Computer Communications, 2005. 29: p. 462-478
- [17] A. T. Tai, K.S.T., William H. Sanders. *Cluster-Based Failure Detection Service for Large-Scale Ad Hoc Wireless Network Applications in Dependable Systems and Networks DSN ' 04*. 2004.
- [18] B. Krishnamachari and S. Iyengar, "Distributed Bayesian Algorithms for Fault-Tolerant Event Region Detection in Wireless Sensor Network", IEEE Transactions on Computers, 2004, 53(3): p.241-250
- [19] Y. Xu, J. Heidemann, and D. Estrin, "Geography-informed energy conservation for ad hoc routing", In Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking, pages 70--84, Rome, Italy, July 2001.
- [20] W. L. Lee, A. Datta, and R. Cardell-Oliver, "WinMS: Wireless Sensor Network-Management System, An Adaptive Policy-Based Management for Wireless Sensor Networks," School of Computer Science & Software Engineering, Univ. of Western Australia, tech. rep. UWA-CSSE-06-001, 2006.
- [21] M. Younis, M. Youssef and K. Arisha, "Energy-Aware Routing in Cluster-Based Sensor Networks," Proceedings of the 10th IEEE/ACM International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS2002).
- [22] M. Ding, D. Chen, K. Xing, and X. Cheng, "Localized fault-tolerant event boundary detection in sensor networks", in Proceedings of the 24th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '05), vol. 2, pp. 902-913, Miami, Fla, USA, March 2005
- [23] J.L Chen , H.F Lu and C.A Lee, Autonomic self-organization architecture for wireless sensor communications, International Journal of Network Management, v.17 n.3, p.197-208, June 2007