A HYBRID MODEL FOR ENHANCING MANAGEMENT AND PERFORMANCE OF WIRELESS SENSOR NETWORKS

By

Ziad Azzi

A Thesis Submitted
In partial Fulfillment of the Requirements for the Degree of
Master of Science in Computer Information System
Department of Computer Science

Faculty of Natural and Applied Sciences
Notre Dame University – Louaize
Zouk Mosbeh, Lebanon
Fall 2005

A HYBRID MODEL FOR ENHANCING MANAGEMENT AND PERFORMANCE OF WIRELESS SENSOR NETWORKS

By

Ziad Azzi

Committee Members:

H Maalouf

Dr. Hoda Maalouf: Assistant Professor of Computer Science and Chairperson.

Advisor.

Dr. Khaldoun El Khaldi: Assistant Professor of Computer Science.

Member of Committee.

Dr. Mario Missakian: Assistant Professor of Computer Science.

Member of Committee.

Ziad Rached

Dr. Ziad Rached: Assistant Professor of Mathematics.

Member of Committee.

Date of thesis defense: **February 27, 2006**

Acknowledgements

This thesis is by far the most significant scientific accomplishment of my life and it would be impossible without people who have supported me and believed in me.

Most of all I would like to gratefully acknowledge the enthusiastic supervision of Dr. Hoda Maalouf during this work; I cannot imagine having a better advisor for my thesis, and without her knowledge and perceptiveness I would never have finished.

I would also like to thank my parents for their love and encouragement and for creating an environment in which following this path seemed so natural.

I am indebted to all my friends at the Notre Dame University, and to the many members of the staff at the Computer Center who encouraged me on this journey.

And last but not least for someone special, not necessarily for coming along at the right time, but for the very special person she is. And for the incredible amount of patience she had with me in the last six months.

Abstract

Wireless sensor networking is an increasingly important technology that is used in a variety of applications, such as environmental monitoring, infrastructure management, public safety, medical, home and office security, transportation and military systems. However, until now, wireless sensor networks (WSNs) and their applications have been developed without considering an integrated management solution.

This thesis analyzes in detail emerging wireless sensor management schemes and proposes a hybrid network model that establishes a combination between event-driven and continuous WSNs.

The main reasons behind the adoption of a hybrid strategy are to enhance WSNs' management schemes, to promote the productivity of the network resources and to solve energy management problems.

The proposed strategy divides the network into two stages in which the first stage is an event-driven network and the second is a continuous one. Cluster heads were used as an interface between the two stages and periodically transmitted their data, depending on their availability, to the sink node.

Also, cluster heads played major roles WSN's management functional areas, especially in configuration, performance and accounting management. In this thesis, we build a simulation model to analyze different performance parameters of the proposed system and to find an optimum value *T* for cluster heads period.

Table of Contents

List of Figures

Chapter 1

Introduction and Problem Definition

1.1 Introduction

Wireless sensor networks (WSNs) consisting of a large number of sensor nodes deployed over an area and integrated to collaborate through a wireless network, encourage several novel and existing applications, such as environmental monitoring, health care, infrastructure management, public safety and military. These applications were enabled by the rapid convergence of three technologies namely digital circuitry, wireless communication, and micro electro mechanical systems. In fact, advances in these technologies have led to very compact and autonomous sensor nodes, each containing one or more sensor devices, computations and communication capabilities, and power supply. The physical dimensions of sensor nodes tend to be small (e.g., 3cm or 3mm) and the size limitation ends up restraining the power supply capacity and computational resources of the sensor nodes.

Wireless sensor networks promise several advantages over traditional sensing methods: better coverage, higher resolution, fault tolerance and robustness. The ad hoc nature and deployment make them even more attractive in military applications and other risk-associated applications such as disaster and toxic zones [10, 2].

Until now WSNs have grown in popularity and this trend will increase in the foreseeable future. Developing middleware software that allows devices to form efficient transient structures by discovering each other, and sharing and trading their constrained resources dynamically will be very important. Enabling self-management and self-organization of the mobile devices is needed because ad hoc wireless networks are not fully manageable from a central location. It is therefore challenging to provide efficient monitoring of such systems due to their highly dynamic nature, decentralized operation and often disrupted connectivity.

Managing wireless networks is a significantly harder task than managing wired networks for many reasons. One of the main problems is the unpredictable behavior of the wireless channel, due to fading, jamming and atmospheric conditions. Signal quality can vary quite dramatically, which might suddenly reduce the efficiency of the management operation. The bandwidth of wireless links is another issue that will always be limited due to the properties of the physical medium and regulatory limits on the use of the radio spectrum. Therefore, it is necessary for wireless network protocols to efficiently utilize the available bandwidth.

There are a number of interesting studies about network management using mobile agent technology [16]. Using mobile agents has some benefits, like reduction in network traffic, efficient utilization of computational resources, support for heterogeneous environments and increased flexibility. On the other hand, use of mobile agents absorbs considerable resources from the agent host which can be an essential problem for mobile terminals.

WSNs pose numerous new challenges. An obvious one is management of such networks. It is clear that the traditional management paradigms will not scale, and the emphasis should be on self-management, autonomous intelligence, self-organization and in-core data processing.

1.2 Problem Definition

The basic architecture of a wireless sensor network consists of sensor nodes connected to each other wirelessly. The flow of data usually traverses the network from the source sensor nodes to a destination sensor node (i.e. the sink).

An event-driven WSN is one which reports data to the observer only when certain events occur, as opposed to continuous networks, which report data at regular intervals. To the best of our knowledge, little research has been done on failure detection in WSNs and even though some proposals exist, their focus is on continuous networks. Event-driven networks pose special challenges to the problem. In fact, in a continuous system if a sensor fails and stops sending its data, the sink node will easily detect the node failure and act accordingly. However, this problem can go undetected in an event-driven system because the sink would relate the absence of incoming data to no events happening.

In addition to the problems mentioned earlier for wireless networks, namely the unpredictable behavior of the wireless channel and the limited available bandwidth, WSNs have a major problem which is energy consumption.

In fact, energy management in WSNs is crucial since battery-driven sensor nodes are severely energy constrained. One main problem that affects WSN management and in particular the residual energy scan, is that it is not fault-tolerant to a node failure during a scan. Suppose a node malfunctions during a search. The data sent by the node's descendants will not be forwarded, and these nodes will become orphans. Hence the need to find a management scheme that is capable of dealing with these issues to guarantee that the network quality of service detects failures and recovers from them.

1.3 – Research Objectives

This research concentrates on the analysis of wireless sensor network management and has the following objectives:

- To remedy the above mentioned problems that face WSNs by implementing a hybrid management scheme that divides a one-stage WSN into two stages, in which the first stage is event driven and the second one is continuous. The hybrid (event driven *and* continuous) WSN is homogeneous and hierarchical. The sensor nodes only disseminate the data when an event in the monitored area occurred in the first stage and when cluster controllers work as interfaces between the event-driven network parts and the continuous network parts periodically.
- To identify how cluster controllers can be used to enhance the management scheme of the whole system.

1.4 Approach

First, we start by comparing the two sensing strategies for one-stage WSNs. Then we divide the network into two stages using cluster head sensors (or cluster controllers) and analyze the effect of periodic transmission at cluster controllers on the system management performance. Here we aim to propose and evaluate the performance of a hybrid management architecture for WSNs. In fact, we took a deep look at fault and performance management capabilities of event-driven and continuous WSNs and then we showed how the use of automatic management services at cluster controllers can provide self-configuration, self-diagnosis, and self-healing (some of the self-managing capabilities). We also showed that the proposed management solution minimizes energy consumption without incurring a high cost to the network response time.

1.5 Thesis Organization

This thesis concentrates on emerging wireless sensor management, covering management issues and technologies. It proposes a hybrid network management and builds a simulation model to prove the efficiency of the suggested system.

The thesis is structured into five chapters. In addition to the present one, chapter 2 gives an overview of WSNs, their characteristics and potential applications. Chapter 3 focuses on WSN management. Chapter 4 introduces the hybrid management model for wireless sensor networks and analyzes in detail the concatenation of event-driven wireless sensor networks to continuous wireless sensor networks. Also in this chapter, some simulations were carried out to show how the power management problem can be solved. Finally chapter 5 includes summaries of the main results and suggests some potential future work.

Chapter 2

Overview of Wireless Sensor Networks

2.1 Background

Recent advances in wireless communications and electronics have enabled the development of low-cost, low-power, multifunctional sensor nodes that are small and can communicate over short distances.

The increasing sophistication of monitoring and controlling systems with multiple sensors has generated a great deal of interest in the development of wireless sensor networks (WSN). They provide distributed network access to sensors, actuators and processors embedded in a variety of equipment, facilities, and environments, representing a significant improvement over traditional sensors. WSNs aim to collect data and sometimes control an environment. This kind of network may consist of hundreds to thousands of sensor nodes that have the capability of sensing, processing and communicating using a wireless medium [3, 4].

Figure 2.1 below illustrates a WSN life-cycle phase called “network self-boot up.”

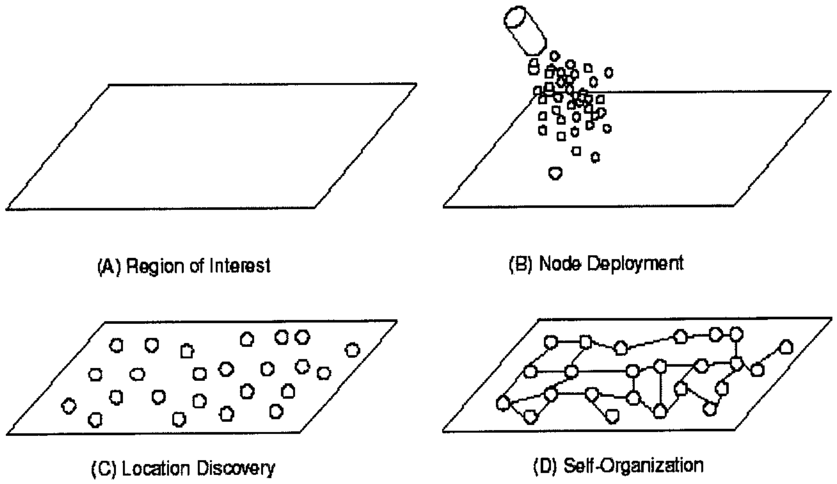


Figure 2.1: WSN boot up

A sensor network is composed of a large number of sensor nodes that are densely deployed either inside the phenomenon or very close to it. The position of sensor nodes need not be engineered or predetermined. This allows random deployment in inaccessible terrain or disaster relief operations.

The sensor nodes are deployed over an area (Figure 2.1B). They are able to discover their locations (Figure 2.1C) and organize themselves as a wireless network (Figure 2.1D). The node deployment can be done, for example, by dropping a large number of sensor nodes from an airplane in a certain area or placing them in this area by hand or robotically. A WSN must be able to operate under very dynamic conditions and work unattended in remote areas.

Once the network is formed and the sensor nodes are operating, most sensor nodes will be able to sustain a steady state of operation i.e., their energy reservoirs will be nearly full, and they will be able support all the sensing, processing and communication tasks required. In this mode, sensor nodes will constitute a multi-hop network. The sensor nodes begin to establish routes by which information is passed to one or more sink nodes (Figure 2.2).

2.1.1 Sink Node

Sink nodes are typical sensor nodes that usually differ from other types of sensor nodes in the following aspects: they have more energy, longer radio range and do not perform sensing. Furthermore, the sink node may be a data gatherer mobile node.

In the literature, sink nodes are also called *monitoring nodes* [10]. Any other entity (non-node) required to perform the functionalities of a sink node is usually called a b*ase station* (BS) .The main difference between a sink node and a base station is that a base station has no resource limitation. Sink nodes and BSs can serve the purpose of collecting information from the WSN and sending it to one or more external entities called *observers*.

2.1.2 Naming of Sensor Nodes

Because sensor data are intrinsically associated with the physical context of the phenomenon being sensed, spatial coordinates are often a natural way to name data [17]. Besides the addressing (naming) purpose, the node location can be employed by routing protocols that use spatial addresses and by signal-processing algorithms (e.g., beamforming) that are used for tasks such as target tracking. In some applications, the resource constraints of WSN can be better met by an attribute-based naming system than by traditional approaches such as IP-addressing. Application-dependent systems such as WSN can name and route data directly in application-level terms [14]. Thus, in some cases sensor nodes may not have global identification (ID) because of large overhead and a large number of sensor nodes.

2.1.3 Multi-hop Network

In the case of ad hoc deployment, the sensor nodes should be able to cope with the resultant distribution and form connections among them. The limited available energy and small form of the sensor nodes limit the radio transmission range and suggest small multi-hop transmission schemes (Figure 2.2).

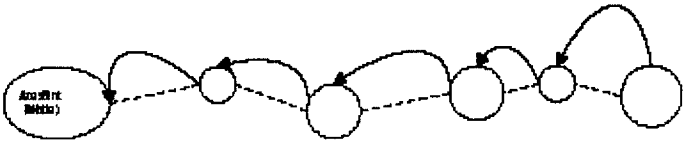


Figure 2.2: Multi-hop communication

Although the multi-hop network can operate in both sensor-to-sink or sink-to-sensor (broadcast or multicast) modes, the bulk of traffic will happen in sensor nodes near the sink node. This is due to the fact that disseminated data from all source nodes to the sink node use intermediate nodes, putting a significant strain on the energy resource of the sensor nodes near the sink and making that neighborhood more susceptible to energy depletion and failure. This situation is called the *energy wave problem*. Figure 2.3 illustrates the energy wave problem which occurs due to the multi-hop communication scheme. The dark region represents unavailable sensor nodes due to the energy wave problem. However, sensor nodes may fail due to other reasons such as mechanical failure [16].

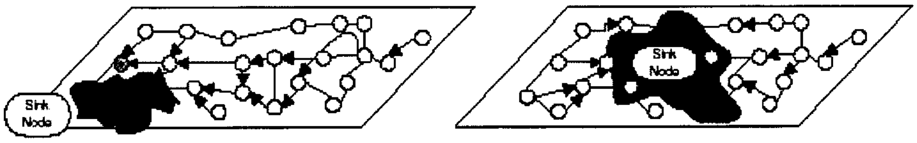


Figure 2.3: Energy Wave problem

2.1.4 Node Failure and Fault Tolerance

When sensor nodes fail, the medium access control (MAC) and routing protocols must accommodate the formation of new links and routes to the sink nodes. This may require rerouting packets through regions of the network where sensor nodes have more energy left [4]. In the cases illustrated in Figure 2.3 the network is partitioned and the sink nodes become isolated. The failure of sensor nodes should not affect the overall task of the WSNs. This property is called *fault tolerance* i.e. the ability to sustain network functionalities without any interruption despite sensor node failures.

2.1.5 Clustering

Communication is the major energy consumer in wireless networks, especially data transmission. The transmission power required by communication between nodes depends on distance. To reduce the amount of power spent on long distance radio transmission and to minimize the energy wave problem, the sensor nodes can also be clustered [9].

The clustering algorithms can include cluster-head (leader) election mechanisms such that each sensor node is associated with a cluster-head as its leader. The cluster-head (common-node) relationships are established between sensors that are able to communicate with each other. Communication between common-nodes and cluster-heads can be multi-hop (Figure 2.4 A) or single-hop (Figure 2.4 B).

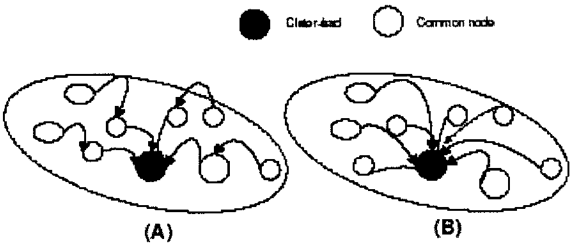


Figure 2.4: Cluster communication scheme

WSN and sensor node architecture are completely dependent on the purpose of the application. The section below presents the main components of sensor nodes that can be applied to WSNs.

2.2 Wireless Sensor Node Architecture

A wireless sensor node is composed of four basic components: a *power supply*, a *computational module* (processor and memory), a *transceiver* and a *sensor unit* (Figure 2.5). The physical and logical components of a wireless sensor node are presented below.

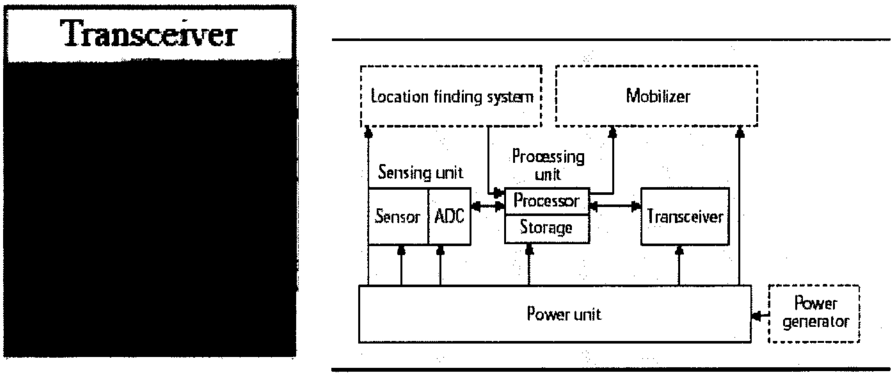


Figure 2.5: Components of a sensor node [8].

Power Supply. The most widely used power supply in sensor nodes is the battery. The choice of the battery type is important since it can affect the design of the sensor node [2].

Computational Module. Composed of a processor and a memory device, this module permits the sensor node to process local data. Developing a node for ultra-low power represents a critical challenge. In the case of processors, *low power* is the quality of a device that consumes low energy per clock. A device that consumes low energy per instruction is called *energy efficient* [8].

Transceiver. The transceiver connects the node to the network. The main types of transceivers are: radio frequency (RF), infrared and optical [8].

Sensor Unit. The sensor unit can be composed of one or a group of sensors which are devices that produce an electrical response to a change in physical conditions. Sensing devices can differ widely in terms of physical characteristics and design, manufacturing, modeling and signal processing. Thus, numerous models of varying complexity can be constructed based on application needs and device features (e.g. pressure, humidity, acceleration, temperature, flow meter, pH, gas, altitude, chemical, biological and medical).

Software. Software is used to represent a set of programs and procedures which becomes an autonomous system capable of performing information processing, relaying and management tasks. As previously seen, wireless sensor nodes have strong hardware and software restrictions on processing power, memory capacity, battery lifetime and communication throughput. Thus, software designed for WSN nodes must consider those limitations.

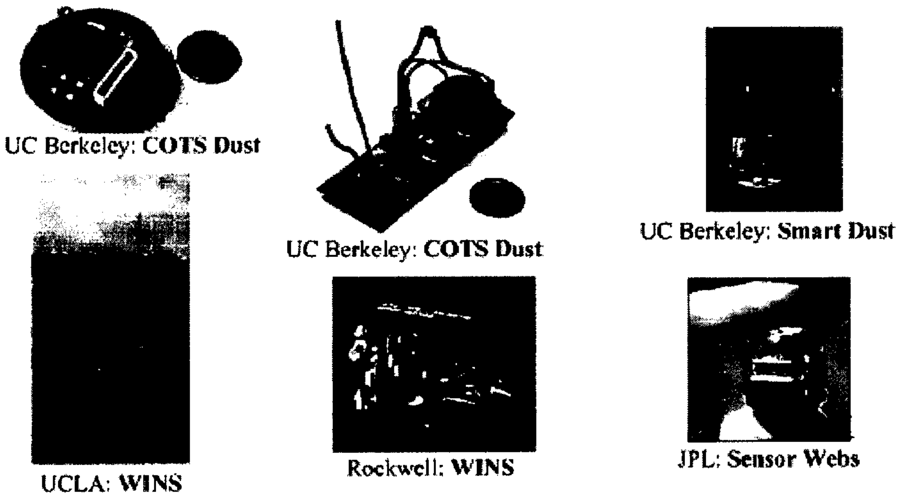


Figure 2.6: Wireless sensor node projects.

A sensor node may have additional dependent components, such as a location-finding system, a power generator and a mobilizer [10]. All of these units are expected to have small dimensions, consume ultra low power, operate in high volumetric densities, have low production cost, be disposable and autonomous, operate unattended and be adaptive to the environment. These design factors have been addressed by many researchers.

Figure 2.6 above shows some wireless sensor nodes such as Smart Dust from University of California at Berkeley [26], WINS (Wireless Integrated Network Sensors) from UCLA [29] and Rockwell and JPL *Sensor Webs* from NASA's Jet Propulsion Lab.

The mobile and static wireless sensor nodes have the ability to gather spatial as well as temporally dense data over vast geographical areas. The cost of a single node is very important to justify the overall cost of the network. In some cases, the network cost is more expensive than that of the traditional wired sensor networks, although the WSN is cost-justified by benefits.

2.3 WSN Applications

Wireless sensor networks have the advantage of spanning a large geographical area and being able to detect and track events collaboratively. A number of high-profile applications for WSNs have been proposed to monitor the environment, detect, classify and locate events, and track targets over a specific region. All WSN parameters can vary depending on the application considered. For example, the deployment can be either (1) predetermined when the environment is sufficiently known and under control, such that the sensors can be strategically hand-placed or (2) undetermined when the environment is unknown or hostile, in which case the sensors may be airdropped from an aircraft or by other means. Regardless of the diversity of applications, the following features are true for all of them: low power, low cost, wireless and ad hoc. Among various applications for WSNs, the most interesting are described next [5, 28].

Disaster area surveillance. Several thousand sensors are thrown from an airplane and rapidly deployed in a disaster area. The sensors communicate and coordinate to form an ad hoc communication network. Emergency response teams can disseminate concurrent queries into the WSNs to collect information in the disaster area.

Civil applications. There are many of varied nature. One example is pollution

detection along beaches with sensor nodes distributed along the shoreline [6]. WSNs can also be spread throughout the exhaust system of an urban area to detect level of air quality.

Intelligent transportation systems. There are wireless sensors developed for counting passing vehicles, measuring the average roadway speed, and detecting ice and water on the road. Clusters of sensors can transmit this information in near real-time to wired base stations for controlling and predicting traffic, and in clearing road hazards [6].

Monitoring forests, volcanos, twisters, and so on. This application is for environment centered sensing that wants the WSN to report the occurrence of a critical event with minimum delay, also providing information about its location.

Tracking the enemy in military applications. In this type of application, a WSN is deployed in a field to monitor movements of enemy tanks which are considered targets. The movement of tanks is detected by the seismic sensor on the sensor nodes and the target detection is propagated back to the control center where the information can be further analyzed by observers (a central server and human operators) [25].

Helping fight against terrorism. WSN researchers are also focusing on systems for detecting and tracking threats. This kind of WSN consists of an easy-to-deploy system with a number of wireless sensors (e.g. seismic, magnetic, pressure, acoustic, nuclear or particle-counting) tied together with a communication network and a scheme for converting the data into forms easily interpreted by users [24].

Human-embedded smart sensor network. Implanted biomedical devices have the potential to revolutionize medicine. Smart sensors, which are created by combining sensing materials with integrated circuitry, are being considered for several biomedical applications, such as glucose level monitors or retina prosthesis [20].

Planetary exploration. WSNs can replace spacecraft that have been orbiting the moon other planets such as Mars. A spacecraft orbiting Mars has detected large quantities of water-ice just below the surface of the planet. The American space agency NASA has invested in research about WSNs for interplanetary discovery, called *sensor webs* [15].

2.4 WSN Versus Other Kinds of Networks

A Wireless sensor network differs from other types of networks basically in the following aspects: number of elements, deployments of ad hoc elements, hardware and software restrictions, unattended operations, addressing and routing. Some of these issues will be discussed next. The number of sensor nodes in WSNs can be several orders of magnitude higher than the number of nodes in an ad hoc network.

2.4.1 Sensors Deployment

In general, dense deployment allows greater sensing task and also fault tolerance through a high level of redundancy. Due to deployment of sensor nodes in environments where the nodes may be lost or destroyed, and in cases where sensor nodes cannot be carefully positioned relative to each other and the environment, an alternate strategy to achieve coverage is to deploy greater density of elements [7]. Another motivation for using a large number of sensors is the case in which the incremental cost of deploying a node during initial deployment is much lower than that of deploying new nodes or renewing node resource.

2.4.2 Sensor Energy

Sensor nodes have strong hardware and software restrictions in terms of processing power, memory capacity, battery lifetime and communication throughput. In traditional mobile networks, energy consumption is of secondary importance since batteries can be replaced when necessary. However, in WSNs the main physical restriction is the available energy because batteries are not recharged during the operation of a sensor node in the case of operations in hostile or remote environments and the large number of nodes. All activities performed by the node must take into account energy consumption. As a result, the design of software for wireless sensor nodes must consider these limitations.

2.4.3 Sensor Damage

In computer networks, the replacement of faulty components by technicians is an ordinary operation. The network tends to follow a well-established plan of available resources and the location of each of its elements is well-known. In a WSN this is not often the case, since the nodes are deployed and unattended on an ad hoc basis. In fact, a WSN topology changes very frequently, even if the nodes are stationary after deployment (e.g. nodes thrown into the ocean, in a forest and other remote environments). Dynamic environmental conditions require the network to adapt over time to changing and unpredictable environmental situations. Unattended operation requires automatic configuration and reconfiguration (self-configuration) [7]. Ad hoc deployment requires the system to identify and cope with the resulting topology and connectivity of nodes (self-organization).

2.4.4 Addressing

Inherent to the design of most distributed systems today is the assumption that each node has a unique network address. This address appears in every packet to identify its source and destination. Depending on the WSN application, it may or may not be interesting to identify uniquely each node in the network [13]. The cost of an address in an energy-constrained network can be quite high if the address accounts for a significant portion of the number of bits transmitted. A common alternative in WSNs is to use attribute-based naming. Data is named by attributes and applications request data matching a certain attribute value. Furthermore, observers may be interested in a value associated with a given region and not with a particular node.

2.4.5 Data Flow

In most WSNs, data flow is predominantly unidirectional that is, data flows from source nodes to the sink node (monitoring node or base station). Sensor nodes usually do not have a direct communication channel to sink nodes, which demands that intermediate nodes act as relays to send messages. In this architecture, each sensor node is also a potential relay. The links can be formed by radio, infrared or optical media. The protocol stack must combine power and routing awareness, and promote cooperative efforts among sensor nodes [10].

WSNs are heavily dependent on the purpose of the application. They are employed in specialized tasks and their nodes cooperate among themselves to perform a huge task.

2.5 Conclusion

This chapter shows that the WSNs present many and drastically different challenges. The number of sensor nodes in WSNs can be several orders of magnitude higher than those in an ad hoc network. Sensor nodes are densely deployed, limited in power, in computational capacities and in memory, and are prone to failure. The topology of WSNs changes very frequently. Sensor nodes may not have global identification (ID) because of the large overhead and number of sensors. The position of sensor nodes cannot be engineered or predetermined. This allows random deployment in inaccessible terrain or disaster-related operations. On the other hand, this means that sensor network protocols and algorithms must have self-organizing capabilities. Another unique feature of WSNs is the cooperative effort among sensor nodes. Although many protocols and algorithms have been proposed for traditional wireless ad hoc networks, they are inadequate to the unique features and application requirements of WSNs.

Chapter 3

Wireless Sensor Network Management

3.1 Introduction

This chapter focuses on the problem of managing wireless sensor networks and discusses an organization that can be used in the design of management solutions for different types of WSNs. Managing WSNs is a significantly harder task than managing other networks because of the reasons stated in chapter 2. All of these distinguishing characteristics of WSN can potentially affect the management solution design. In fact, the management of large networks requires powerful abstractions, which permit the identification of management functions on different levels.

3.2 Management Requirements

In a WSN, network element failures are common, nodes can be destroyed, or energy, security, calibration and communication faults may occur.

The network is planned to operate without attention. Nodes can be discarded, lost, and out of operation temporarily or permanently. In this scenario, faults are a common fact, which is not expected in a traditional network. In unpredictable situations, a configuration error (e.g. planning error) may cause the loss of the entire network even before it starts to operate.

WSNs and sensor node architectures are completely application dependent. Thus, the management solution should be “compatible” with the type of application being managed. While depending on the WSN application, it may be interesting to uniquely identify each node in the network. Furthermore, we may be interested in a value associated with a given region and not a particular node; for instance, the temperature at the top of a mountain. A WSN is typically data-centric, which is not common in traditional networks. The objective of a WSN is to monitor and, eventually, control a remote environment.

The main objective of WSN management is to define a set of functions that seek to ameliorate productivity, as well as to integrate in an organized way functions of configuration, operation, administration and maintenance of all elements and services of a sensor network. Thus, all operations performed in the network should be energy-efficient, including the management tasks.

Next, we will discuss the self-managing paradigm for WSNs. Then we will present the management functional areas and management levels as defined for traditional management [1, 18].

3.2.1 WSNs' Self-Management Paradigm

For the reasons discussed earlier, we propose in this thesis that a wireless sensor network must be self-managed and hence must have the following characteristics:

- It has to be responsible for configuring and reconfiguring itself under varying conditions. System configuration (node setup and network boot up) must occur automatically. Moreover, dynamic adjustments are required to handle any changes in the environment or in the network itself.

- It has to look for ways to optimize its functioning and has to monitor its components and fine-tune its workflow to achieve predetermined system goals. The network must be able to discover problems or potential problems, such as uncovered areas, and then find an alternate way of using resources or reconfiguring the system to keep it functioning smoothly. In addition, it must detect, identify and protect itself from various types of attacks in order to maintain the overall system security and integrity. WSN security issues will not be discussed in this thesis.

- It has to know its environment and the context surrounding its activity, and act accordingly. The management entities must find and generate rules to perform the best management of the current state of the network.

A self-managed WSN with such characteristics is also known as *an autonomic system*. An autonomic system is an approach to self-managed computing systems with minimum human interference. The processors in autonomic systems use algorithms to determine the most efficient and cost-effective way to distribute tasks and store data.

The major advantage of autonomic WSNs is their robustness to changes in network states while maintaining the quality of service. However, the computational and energy costs of autonomic processes can be very expensive for some WSN architectures, as well as for services and functions performed semi-automatically or manually.

In fact, the task of building and deploying autonomic management systems in environments such as a WSN which has thousands of network elements with particular features and organization is very complex. This task could become even worse due to the physical restrictions of the sensor nodes in particular, energy and bandwidth restrictions. The building of the management application also depends on the type of the application that is being monitored. A good strategy here is to use management dimensions [19] as described next.

3.3 Traditional Management Dimensions

The use of management dimensions forms a good strategy to deal with complex management situations. This is because it decomposes a problem into smaller sub-problems in successive refinement steps and provides a separation between application and management functionalities through management architecture. As a result, the integration of organizational, administrative and maintenance activities for a given network becomes possible [12].

In general, for traditional networks, management aspects are clearly separated from network common activities i.e., from the services they provide to their users. This separation can be achieved by using two traditional management dimensions called *management functional areas* and *management levels* (or layers). Nevertheless, carrying out these two traditional management dimensions will require new approaches based on the WSN characteristics.

The requirements to be satisfied by systems management activities can be categorized into the following functional areas:

- *Fault management* which involves discovering, isolating and fixing problems in the network. This functional area is responsible for ensuring smooth and continued operation of the network.

- *Configuration management* which is responsible for the initialization and shutdown of the network. It also involves maintaining, adding, and updating new network components. Part of the function of configuration involves defining relationships between network entities.

- *Security management* which involves controlling access to network components and information. This component is also responsible for implementing encryption and decryption schemes for secure end-to-end communication.

- *Performance management* which involves collecting network statistics and tuning the network to improve performance.

- *Accounting management* which involves tracking network utilization by various users and groups. This information can be very useful in network configuration and allocation of network resources to the various groups in an organization.

3.3.1 Logical Layers

To deal with the complexity of management, the management functionality along with its associated information can be decomposed into a number of logical layers namely business management, service management, network management and network element management. The architecture that describes this process is called the *logical layered architecture* [12] and consists of the following layers.

The business management layer is responsible for the management of the whole system. This layer has a broad scope, just a part of which is communication management. Business management can be seen as goal setting, rather than goal achieving. For this reason, business management can be better related to strategic and tactical management instead of operational management.

The service management layer is concerned with the management of those aspects that may directly be observed by the network users. These users may be both final users (customers) and other service providers (administrators).

The network management layer helps manage the functions related to the interaction between multiple elements. At the network management level, the internal structure of the network elements is invisible.

The element management layer is used to manage each network element individually. This layer deals with specific management functions and hides these functions from the network management layer.

As we see so far, WSN management must be simple, adequate to network characteristics and efficient in the use of its scarce resources. The use of management functional areas and management levels is not enough because wireless sensor networks are application-specific. Therefore, a management dimension called *WSN functionalities* was proposed [19] and will be described in the next section.

3.4 Dimensions for Wireless Sensor Network Management

WSNs are embedded in applications to monitor the environment and act upon it. Thus, the management application should try to be "compatible" with the type of application being monitored. A management architecture for wireless sensor networks (MANNA) [19] establishes that in order to have a better development of the WSN management services and functions it is necessary to characterize the WSN and establish a specific management dimension.

MANNA architecture introduces the two well-known management dimensions (functional area management and management levels) and WSN functionalities [19]. Five main WSN functionalities are included in this third dimension for the management configuration, sensing, processing, communication and maintenance (Figure 3.1).

Configuration is the first functionality before the network starts sensing the environment, processing and communicating data. Maintenance treats specific characteristics of the WSN applications during the entire lifetime of the network.

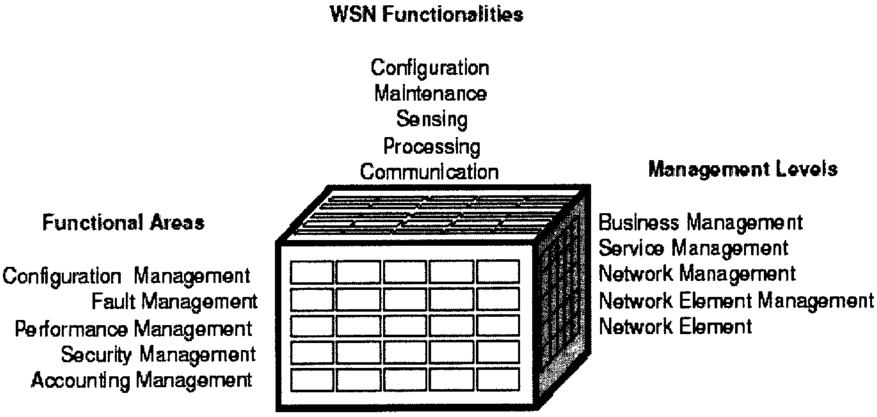


Figure 3.1: Management Dimensions for WSNs.

In this way, the WSN management will have an organization that comes from abstractions offered by the three planes of management functional areas, management levels and WSN functionalities (configuration, sensing, processing, communication and maintenance). The intersection of the three planes (Functional areas, WSN Functionalities and management Levels) defines a cell. Each cell contains a set of management functions. One or more management functions can fit into one or more cells of the cube. Management services are executed through a set of these functions, and the conditions to execute them can be established through the use of policies.

This three-dimensional architecture defines a list of management functions and services that can be performed automatically in order to design self-managed WSNs. In this case, the management services can be executed with little or no human interference. Automatic services and functions can be executed toward self-management if there are appropriate conditions such as a residual energy level.

In the next sections, the WSN management is introduced from the perspective of management level, WSN functionalities, and management functional areas.

3.4.1 Management Levels (or Logical Layered Architecture)

In what follows, we will discuss issues concerning WSN management from the perspective of management levels. As we can see in Figure 3.2 below, the logical layered architecture of a WSN is made of 5 levels.

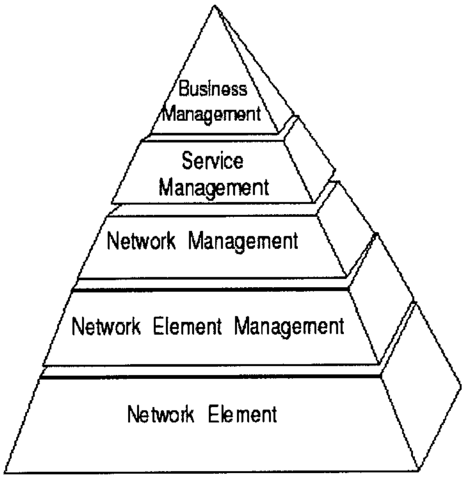


Figure 3.2: Management Levels.

The top level, the *business management level*, deals with the service development and the determination of cost functions. It represents a sensor network as a cost function associated with network set-up, sensing, processing, communication, and maintenance.

Level four is the *service management level.* Wireless sensor network services are concerned with functionalities (Figure 3.1) associated with the application objectives. The basic WSN services are sensing, processing and data dissemination. These services are characterized by a set of parameters which determine their service level. Throughput, response time, packet loss rate and energy consumption are examples of qualifiers of dissemination service.

Level three is the n*etwork management level.* This layer aims to manage a network as a whole, which is typically distributed over a large geographical area. In the network management level, relationships among sensor nodes are to be considered. Nodes can be involved in collaboration, connectivity and aggregation relationships. A WSN must react rapidly to changes in topology, task, degradation and mobility.

The basic functions of a WSN n*etwork element management* (level two) are power management (how the sensor uses its power), mobility management (movement of a sensor node), state management (operational, administrative and usage) and task management (sensing, processing and disseminating schedules). When placed in an environment, the sensor nodes should immediately recognize their own capabilities and functions (self-test) and those of other sensor nodes, and work together as a community system to perform cooperative tasks and networking functionalities. Wireless sensor networks need to be self-organizing, some applications may require networks with a large amount of sensor nodes, and a network element can deal with a single node component or a group of nodes (i.e. cluster of nodes).

Finally, the n*etwork element level* (level one) represents the physical and logical components of a managed element. Physical resources include sensor or actuator nodes such as power supply, processor, memory, sensor device and transceiver. Logical resources include communication protocols, application programs and network services. The most widely used power supply in a WSN is the battery. Since most batteries are not recharged during the operation of a sensor node, the main physical restriction of a WSN is the available energy. All activities performed by the node must take energy consumption into account. Energy consumption patterns of individual nodes and the entire network must be characterized and profiled. This process yields a better understanding of where to apply tradeoffs in the design of the management solution

3.4.2 Wireless Sensor Network Functionalities

This section introduces the third dimension for WSN management [19], which consists of configuration, sensing, processing, communication and maintenance functionalities.

3.4.2.1 Configuration

This functionality is related to the planning, placement, boot up and self-organization of a WSN. The configuration functionality is related to the definition of WSN application requirements, the determination of the monitoring area (shape and dimension), the environment characteristics, the choice of nodes, the definition of the WSN type, and the service provision [27].

An inefficient configuration management may adversely affect the overall performance. WSNs are application-specific, which causes the configuration functionality to change from one WSN to another.

3.4.2.2 Maintenance

The maintenance functionality used in a WSNs can configure, protect, optimize and heal itself without intervention of human operators. Maintenance detects failures or performance degradations, initiates diagnostic procedures and carries out corrective actions in the network. Beyond corrective maintenance, there are other types of maintenance:

- Adaptive, in which the system should adapt itself to meet the changes.
- Preventive, in which the system should learn to anticipate the impact of those changes.
- Proactive, in which it should learn to intervene so as to preempt negative events as the system gets smarter.

Maintenance functionality is needed to keep the network operational and functional, to ensure robust operation in dynamic environments, as well as to optimize the overall performance.

3.4.2.3 Sensing

Sensing functionality depends on the type of phenomenon. Thus, WSNs can be classified in terms of the data gathering required by the application as *continuous*, in which sensor nodes collect data continuously along the time, and reactive in which it answers to an observer's query or gathers data corresponding to specific events occurring in the environment and periodic in which nodes collect data according to conditions defined by the application. The sensing encloses the exposure (time, distance and angle of phenomenon exhibitited at the sensor), calibration and sensing coverage. Depending on the density of the phenomenon, all sensor nodes active all the time may be inefficient [18, 19].

3.4.2.4 Processing

The goal of processing is to reduce the amount of data transferred to the manager of the network management system. In the sensor network management, correlation may be applied to any of the five management functional areas and may be done at several levels of the configuration, from the individual network elements to the maximum level, which involves the entire network. Several types of correlations may be identified, according to the operations performed on the data. The most commonly used methods are data fusion and aggregation [11, 18, 19].

- Data fusion combines one or more data packets received from different sensors to produce a single packet. It helps reducing the amount of data transmitted between the sensor nodes and the observer. Other possible tasks are security processing and data compression.

- Aggregation summarizes current data values in some or all sensor nodes of a WSN. It reduces the amount of data routed through the network, increasing throughput and extending the lifespan of battery-powered sensor networks. Aggregation is essential for wireless sensor networks in which energy resources are limited.

3.4.2.5 Communication

Individual nodes communicate and coordinate among themselves. Two types of communication namely infrastructure and application exist in WSN.

Infrastructure communication is that used to configure, maintain, and optimize operation. The configuration and topology of the sensor network may rapidly change in the case of a hostile environment, a large workload or nodes that fail routinely. Conventional protocols may be inadequate to manage such situations and, thus, new protocols are required to promote WSN productivity.

Application communication (dissemination) relates to the transfer of sensed data or information obtained from it. In order to save energy, short distance transmissions are preferred. Since the sink node (or the BS) may be located far away, the cost to transmit data from a given node to the sink may be high [18, 19].

The communication approach can be classified as follows:

- ❖ Flooding, in which sensors broadcast their information to their neighbors and they in turn broadcast this data until it reaches the observer.
- ❖ Gossiping, in which sensors send data to one randomly selected neighbor
- ❖ Bargaining, in which sensors send data to sensor nodes only if they are interested
- ❖ Unicast, in which sensor can communicate to the sink node, cluster-heads or BS directly.
- ❖ Multi-cast, in which sensors from application-directed groups and use multicast to communicate among group members.

3.4.3 Management Functional Areas

The wireless sensor network management considers that the fault, security, performance and accounting management functional areas are highly dependent on the configuration functional area [19]. In WSNs, all operational, administrative and maintenance characteristics of the network elements, network, services, business and the adequacy performed in the configuration, sensing, processing, communication and maintenance (Figure 3.1) depend on the configuration phase of the WSN. An error in the configuration or a forgotten requisite during the planning may compromise the functionalities of all other areas. This concept is depicted in Figure 3.3 in which the configuration functional area plays a central role. As mentioned before, there are several significant differences between the management of traditional networks and WSNs. In this sense, management functional areas must be rethought considering the WSNs features.

3.4.3.1 Configuration Management

Configuration management is a highly relevant functional area in WSN management. As the objective of a sensor network is to monitor (acquire, process and deliver data) and sometimes to control an environment, any problem or situation not anticipated in the configuration phase can affect the service provided [19].

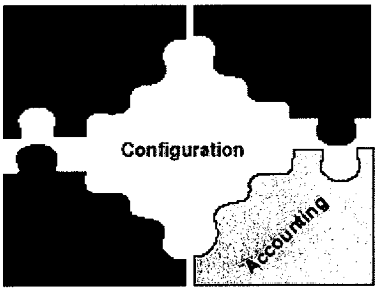


Figure 3.3: The role of configuration management.

The configuration management must provide basic features such as self-organization, self-configuration, self-discovery and self-optimization. Some management functions defined for network-level configuration management are:

- Requirements specification of the network operational environment
- Environmental variations monitoring
- Size and shape definition of the region to be monitored
- Node deployment (random or deterministic)
- Operational network parameters determination
- Topology discovery

- Network connectivity discovery
- Node density controlling
- Synchronization
- Network energy map evaluation

Some management functions that are defined for network element level configuration management are: node programming, node self-test, node location, node operational state, node administrative state node usage state and node energy level.

3.4.3.2 Fault Management

Fault management is a critical function and makes WSN management different from traditional network management. In addition to faults caused by energy problems, other events can happen in a wireless sensor network related to communication, quality of service, data processing, faulty physical equipment environment, integrity and security.

Fault management must provide basic functionalities such as self-maintenance, self-healing and self-protection [18]. Several characteristics of WSNs make us believe that faults are very common in this kind of network. First, large-scale deployment of cheap individual nodes means that node failures from manufacturing defects will not be rare. Second, attacks by adversaries will be likely because these networks will often be embedded in critical applications and deployed in open spaces or enemy territories. Hence in most applications, fault detection is vital not only for fault tolerance of the WSN but also for its security.

3.4.3.3 Performance Management

The challenge in performance management is to perform this task without adversely consuming network resources. There is a tradeoff to be considered: the higher the number of managed parameters, the higher the number of transmitted messages, the higher the energy consumption and the lower the lifetime of the network. On the other hand, if too few parameter values are obtained, it may not be possible to manage the network appropriately [18].

The configuration (in terms of number of sensors, density, node distribution and data dissemination) plays a significant role in determining the network performance. Performance management must consider the self-service characteristic. The performance of the network is best measured with parameters such as system response time, consumed energy and the reliability or accuracy of the results [19]. To the observer, it is likely that multiple samples may be received from different sensor nodes with different data quality. Thus, additional performance metrics exist for example, the *goodput* which is defined as the ratio of the total number of packets received by the observer to the total number of packets sent by all sensors over a period of time and the produced data quality.

Regardless of the application, certain critical features can determine the efficiency and effectiveness of a sensor network [21]. These features can be categorized into quantitative and qualitative features. *Quantitative features* include network settle time, network joins time, network departs time, network recovery time, frequency of updates (overhead), memory requirements and network scalability. *Qualitative features*, on the other hand, include knowledge of nodal location, topology changes effects, adaptation to radio communication environment, power consciousness, single or multi-channel and network security preservation.

3.4.3.4 Security Management

Security functionalities for WSNs are inherently difficult to be provided because of their ad hoc organization, intermittent connectivity, wireless communication and resource limitations. A wireless sensor network is subject to different safety threats: internal, external, accidental and malicious. As a result, information or resources can be destroyed, modified, stolen, removed or lost, and hence, WSN services can be interrupted. Security management must ensure self-protection of the WSN in other words it must provide confidentiality, reliability, disposability, privacy, authenticity and integrity [19].

3.4.3.5 Accounting Management

Accounting management includes functions pertaining to the use of WSN resources and corresponding update reports. These functions can trace the behavior of the network and even summarize the behavior of a given sensor. In a WSN, there is an energy producer (the battery) and some energy consumers (the transceiver, computation module and sensing devices). Operations of the application or management can be measured in terms of energy consumption. Some functions related to accounting management are discovery, counting, storing, and parameter data reporting; network inventory; determination of communication costs determination and energy consumption and traffic checking.

3.5 Conclusion

This chapter discussed the characteristics of self-management for wireless sensor networks. It emphasized that WSN management must be autonomic i.e, self-managed (self-organizing, self-healing, self-optimizing, self-protecting, self-sustaining and self-diagnostic), with a minimum of human interference, and robust to changes in the network states while maintaining the quality of the services.

This chapter also discussed the management challenges for WSNs and explained the three-dimensional WSN management model proposed by [19] which is composed of management levels, management functional areas and WSN functionalities. All three dimensions were explored from a WSN perspective. The management functional areas and the management levels were rethought considering the particular characteristics of WSNs and some management functions are given in each management functional area.

In the next chapter, we consider the third dimension proposed by [19] in greater detail. In fact, we study WSN functionalities (mainly in configuration, sensing, processing and communications) and concentrate on fault management and performance management issues.

Chapter 4

Hybrid WSN Models

4.1 Introduction

In this chapter we consider the different functionalities of event-driven and continuous wireless sensor networks and then propose the use of a hybrid model that enhances WSN management. An *event- driven WSN* is one that reports data to the observer only when certain events occur. On the other hand, *continuous networks* report data at regular intervals.

The cost of sending data continuously may lead to a faster consumption of scarce network resources consequently shortening the network lifetime.

Critical to any wireless sensor network deployment is the expected lifetime. The goal of many WSNs is to have nodes placed out in the field, unattended, for months or even years. Hence, the primary limiting factor for the lifetime of a sensor network is the energy supply. Each node must be designed to manage its local supply of energy in order to maximize total network lifetime.

Also, since a WSN can be deployed into a difficult area (such as a volcano crater, a military field or an ocean), and its number of nodes can be high, recharging or replacing battery nodes may be inconvenient. So, the development of power-saving protocols such as event-driven protocols for the organization of these networks can extend their lifetime. As mentioned previously in WSNs there is a trade-off in performance management to be considered: the higher the number of data packets transmitted (application and management data), the higher the energy consumption and the lower the network lifetime. On the other hand, if parameter values are not obtained (by minimizing the number of packets sent), it may be hard or impossible to manage the network appropriately. Hence a major challenge in this thesis is to find a WSN model that can perform its management tasks without adversely consuming network resources. In other words, we aim to find a tradeoff among energy consumption, latency (delay), and quality of management service (message delivery, accuracy, etc).

This chapter is comprised of five sections. Section 2 describes the different types of WSNs. Section 3 presents the proposed hybrid WSN. Section 4 describes the built simulation model and section 5 explains the main simulation results found in this research.

4.2 Different Types of Wireless Sensor Networks

Given that any type of management is highly dependent on the configuration of the network of interest, WSNs can be classified as follows: A WSN is *homogeneous* when all nodes have the same hardware capabilities (processor, memory, battery and communication device features). When the WSN is comprised of nodes with different capabilities it is said to be *heterogeneous*. A WSN is *hierarchical* when the nodes are organized into groups which themselves can be organized into different hierarchies. Each group has its own leader and belongs to a hierarchical level. When the nodes are organized in only one level, the hierarchy of the groups is one. A WSN is said to be *flat* when its nodes are not organized into groups [11].

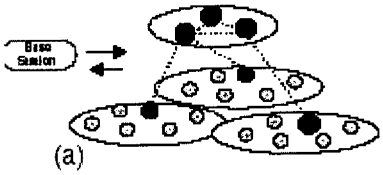


Figure 4.1: Hierarchical WSN.

In heterogeneous hierarchical networks, the nodes with the highest capabilities can assume leadership during the entire network lifetime.

In homogeneous hierarchical networks, when the leaders receive the information from the common nodes, they can perform some processing (such as fusion and aggregation) and disseminate the resulting information using multi-hop communication to the base station.

Basically there are three different types of sensor nodes:

- *Common-nodes* which are responsible for collecting sensing data.
- *Sink nodes (monitoring nodes)* which are responsible for receiving, storing and processing data from common-nodes.

- *Cluster heads:* which are the leaders of group responsible for receiving, storing, and processing data cluster nodes.

4.3 Proposed Hybrid Model

In this work, we focus on event-driven WSNs and how they can be modified and improved. As explained earlier, an event-driven WSN is one that reports data to the observer only when certain events occur. To the best of our knowledge, little research has been done on fault management and performance management WSNs and even though proposals do exist, their focus is on continuous networks. Event-driven networks pose special challenges to the problem.

Most extant research focuses on continuous WSNs. Under normal conditions, the sink in a continuous WSN receives sensing data (traditionally called *SENSOR-REPORT MESSAGES*) at regular intervals. This stream of data not only delivers the content the end-user (i.e. sink) is interested in, but it also works as an indication of how well the network is operating. When the management application receives data from every single node, the sink knows that everything is doing fine. If, however, the management application stops receiving messages (SENSOR-REPORT) from part of the entire network, the observer knows that a failure has occurred. This is not the case with event-driven WSNs.

Also, in continuous WSNs the redundant nodes produce redundant data, generating traffic and leading to collisions and loss of energy. In this kind of wireless sensors network, energy management is probably the main aspect to be considered, since the WSN lifetime depends on its rational use.

However an event-driven WSN has characteristics that differ from continuous WSNs. When observers do not receive any information from the network, they may suppose that no event has happened. However, in some cases, the network or part of it may be unavailable due to an energy problem or other types of failures or attacks. So event-driven protocols save energy (due to minimizing the number of messages transmitted) but, in terms of failure detection, event-driven networks present challenges that are not faced by continuous networks. In an event-driven WSN without management, when the observer does not receive any data, it supposes that no event has happened.

However, this may not be the case because the nodes can be unavailable or out of service for different reasons.

Due to all the above mentioned problems facing continuous and event-driven WSNs, we propose the following architecture for wireless sensor networks:

A hierarchical WSN that is hybrid, with the sensor nodes divided into several clusters, and one sensor elected as a cluster head. The communication between the sensor nodes and the cluster head is based on an event-driven protocol.

We also propose that the cluster heads will report continuously to the base station (or sink). So, the proposed hybrid model is made of the concatenation of an event-driven first stage and a continuous second stage. In this proposal model, cluster heads will play major roles in the management of the WSN, especially when dealing with fault management and performance management. In fact, cluster heads will periodically report to the base station about the status of its cluster. This will enhance the overall management of the WSN.

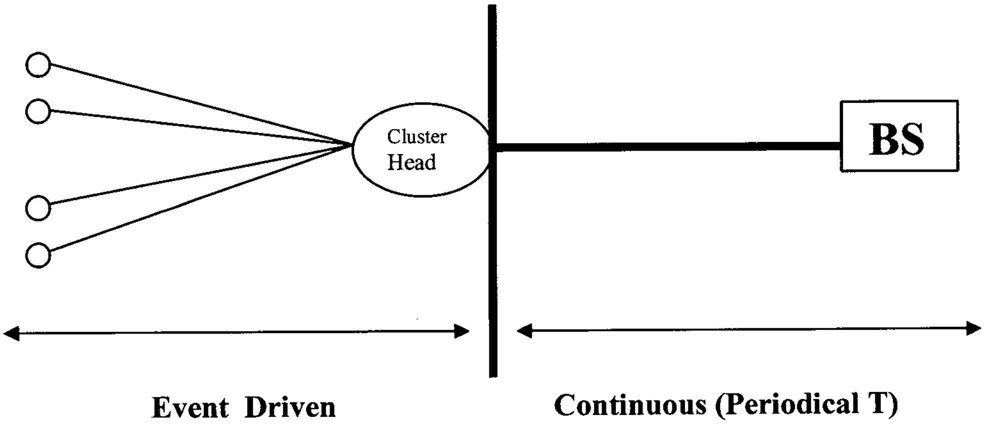


Figure 4.2: Hybrid WSN

Although most researchers assume that cluster heads are different from cluster sensor nodes, we do not make that assumption because in reality all nodes are similar and can switch jobs at any time. This would be required when a cluster head fails and the cluster will have to choose a new leader.

We should note that in order to have a single-hop communication between the cluster head and the base station in the continuous second stage, it is necessary for the elected cluster head to change its radio configuration, increasing its range and, consequently, the energy consumption. Therefore, cluster heads consume more energy than common nodes and have a shorter lifetime.

Now that we have analyzed some functionalities of the hybrid model, namely the configuration, the sensing and the communication we must specify how the processing is done at the cluster heads. Aggregation will be implemented at these nodes. In fact, a cluster head will periodically transmit the data it collected so far. This concept is applied both for application data and management data.

Now that we have explained in detail the proposed hybrid model, a major issue here is to find an optimum value of "*T*" the reporting period between cluster heads and the base station. Finding this optimum value will make the system work optimally (i.e. minimize the delay and the number of messages transmitted, hence minimizing the energy consumption).

For that purpose, a simulation will be built. The following sections present the simulation approach and information, developed as a case study of the use of the hybrid architecture proposed in this thesis. Simulations are performed to show how the management solution can promote the network productivity and to evaluate the impact of the hybrid network model on WSN performance and service.

In order to evaluate the performance of the proposed model, we define a simple event-driven application that runs in the WSN for monitoring the environment temperature. We show that our solution achieves a compromise between system response time and energy consumption.

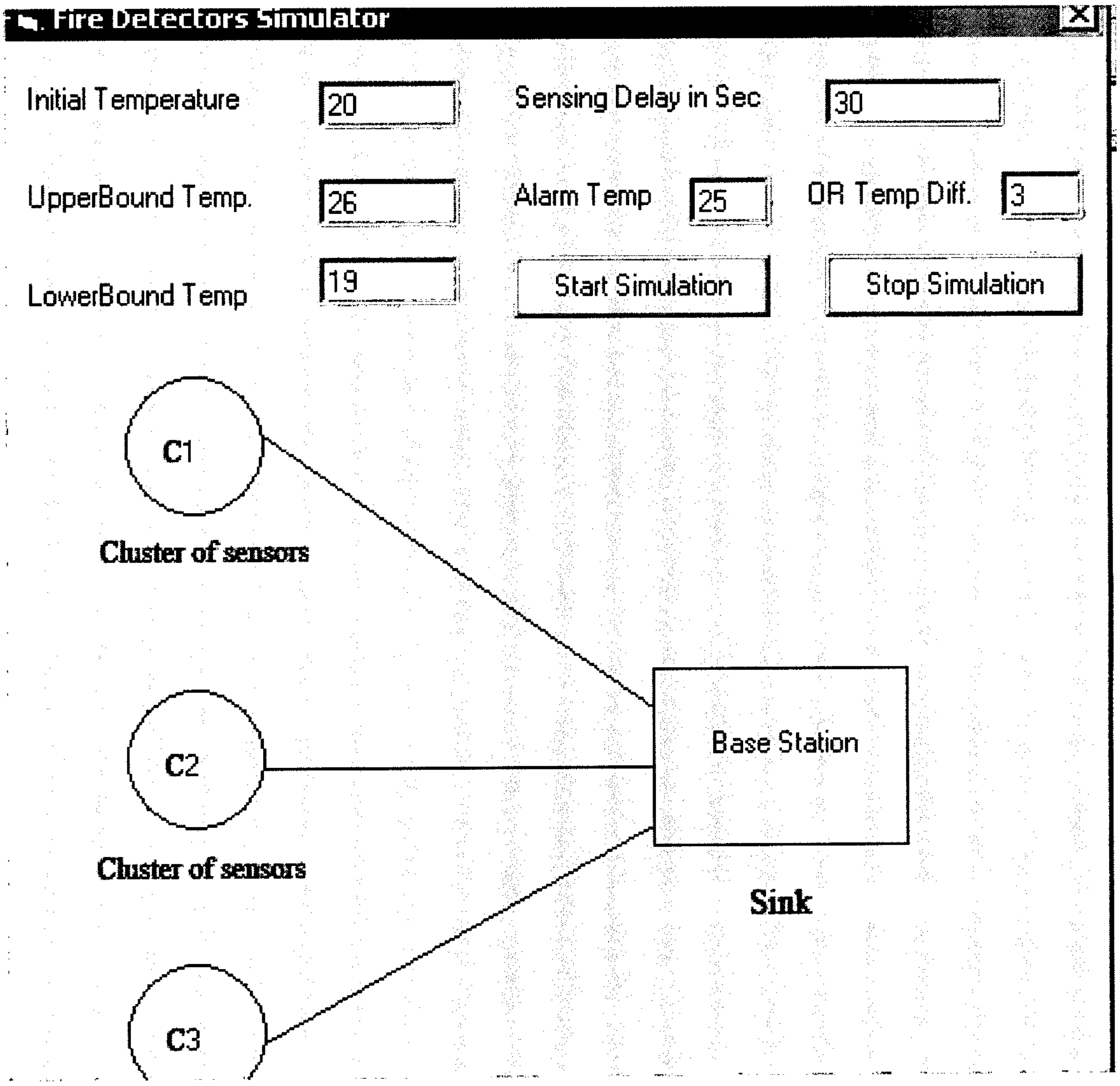
4.4 Simulation Model

Fire-detection systems can help to reduce the damage caused by burning. In fact, the ability to detect and locate a fire quickly and effectively is at the heart of almost all fire-detection systems. Most the fire-detection systems are based on digital image processing, obtained from specific orbital satellites, by finding pixels with a brightness temperature above a certain threshold. On the other hand, fire detection can also be performed by other methods of activation, mainly temperature or smoke. In the case of temperature, the fire-detection system can be set to trigger an alarm at a given temperature or to report temperature rises. In this context, wireless sensor networks can be used as a fire-detection system due to their ability to collect information from the environment using sensors. A sensor network can be programmed to report temperature rises, air humidity and even wind direction. These data are useful to determine the fire probability and help in more efficient fire fighting. In order to model a real application for fire detection, all available information about the monitoring area should be considered.

For our study, we have conducted a set of experiments taking into account distinct simulation scenarios. We have defined a WSN application and some management functions, as mentioned before, and evaluated the performance of the system using a Visual Studio 6.0 script as the Network Simulator. Each scenario was emulated several times.

In our application entitled “Fire-Detector Simulator” the temperature is the monitoring object. Although the nodes sense the temperature continuously along the time, data are sent to the cluster heads only when the new temperature differs from the current temperature by 3 degrees or more from the last data sent, inducing the event-driven property between the sensing nodes and the cluster head. Also the data are sent only if the temperature reaches 25 degrees which is the maximum allowed.

In order to simulate the temperature behavior of the environment, random numbers were generated following a Gaussian distribution, taking into consideration standard deviation of 1 from an average temperature of 20 degrees.



Initial Temperature 20

Sensing Delay in Sec 30

UpperBound Temp. 26

Alarm Temp 25

OR Temp Diff. 3

LowerBound Temp 19

Figure 4.3: Fire-Detector Simulator

Figure 4.3 above describes the main parameters and the features used in the simulation models. As we can see we have the initial temperature, where our simulation uses this temperature at time zero. The upper and lower bound temperatures are the maximum and lower temperature boundaries.

The sensing delay, is the periodic window for sending sensing, the critical sensing (data are sent only) is when the temperature reaches the maximum value allowed by the system which is 25 degrees, and when the new temperature differs from the current temperature by 3 degrees or more.

The available temperature data (in degrees Celsius) can be described as follows:

- Maximum absolute temperature: 26 degrees
- Minimum absolute temperature: 19 degrees
- Initial absolute temperature: 20 degrees

In addition, we assumed the following temperature information about fire detection:

- The minimum temperature to be considered as fire is 25 degrees;
- Temperature variations below 3 degrees are considered normal and do not need to be reported;
- Variations above 3 degrees in a short period of time are considered abnormal and should be analyzed as a possible fire, even if the temperature remains below 25 degrees.

In this simulation, we divide the network into 3 clusters. Figure 4.4 below shows the three cluster heads, C1, C2 and C3. We assume that each cluster contains the same 1 number of sensor nodes (5 nodes). These cluster-heads are connected directly to the base station where this program calculates the number of sensing arrivals to the base station. Cluster-heads transmit their data periodically with a period *"T"*. One major aim of this simulation is to find an optimum value of "*T*". If "*T*" is small, than no reading or few readings will be sent to the BS. However, if "*T*" is large enough, the cluster-head will be able to combine several readings and transmit them to the base station. But, a larger value of "*T*" could increase the system response and jeopardize its performance.

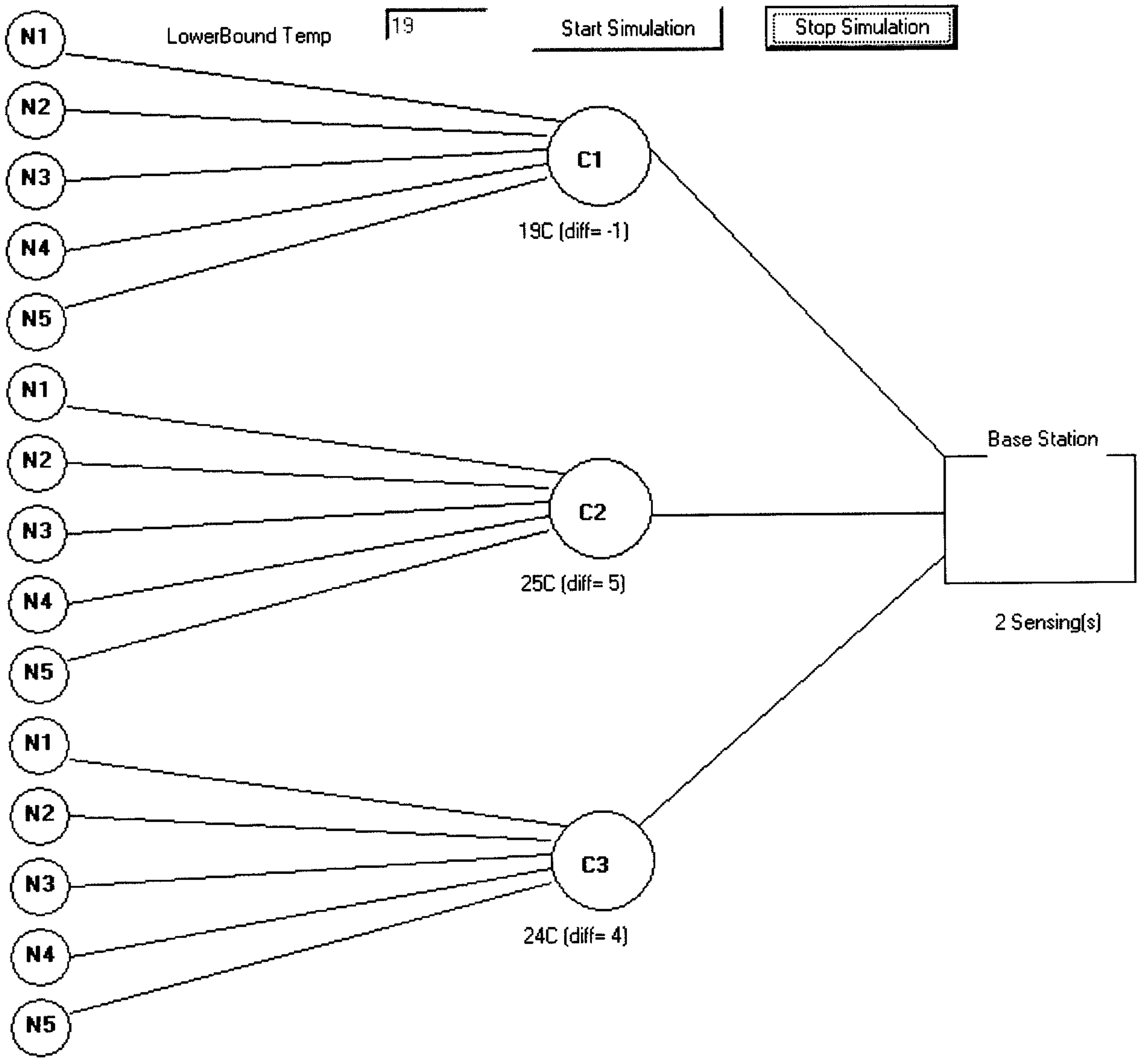
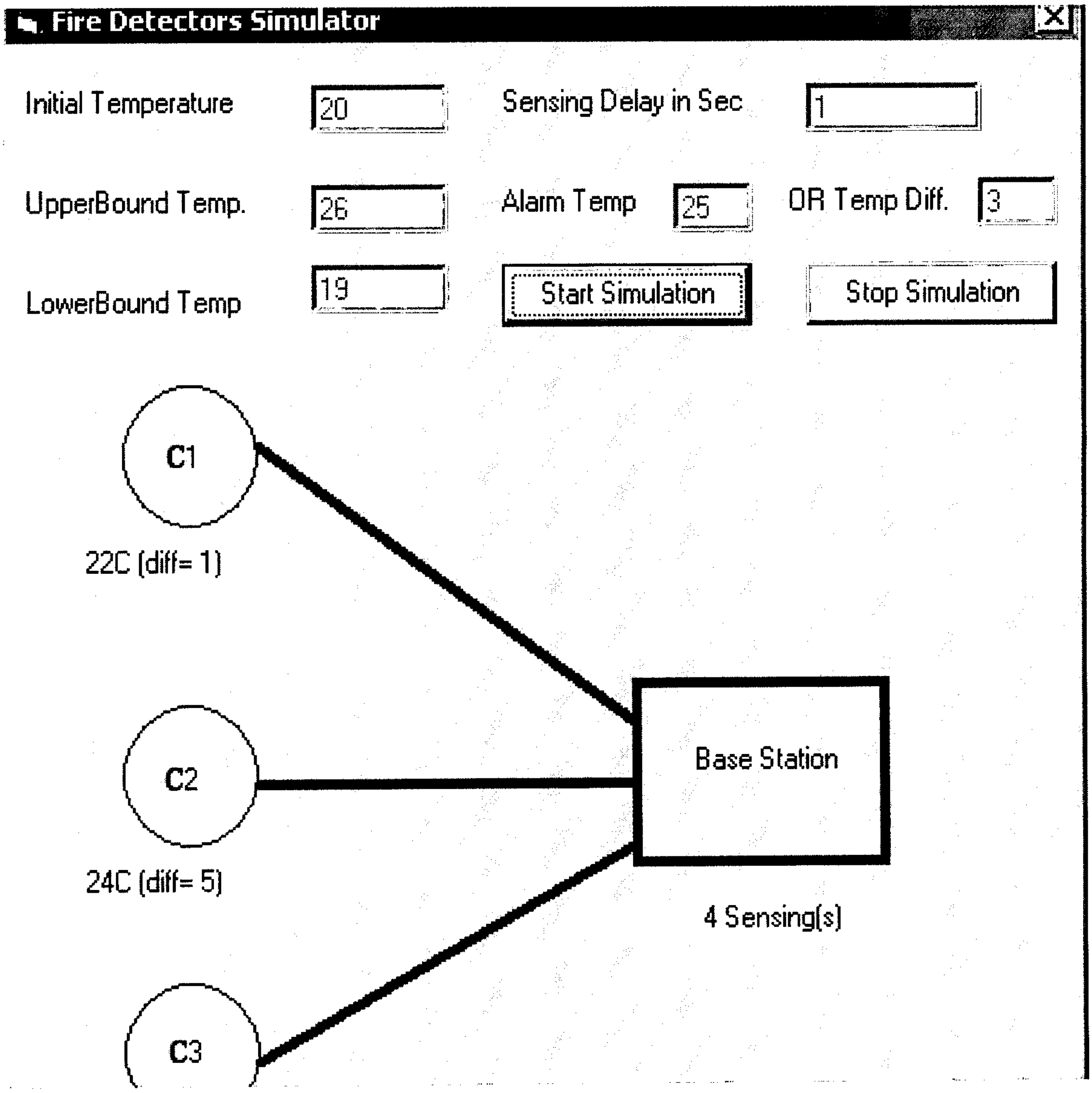


Figure 4.4: Nodes and Cluster Distributions.

At first, and in order to evaluate an estimate of "T", we assume that the cluster head does not store any reading and the readings are sent straight away to the base station (i.e. $T = 0$). In this case, under each cluster "C" we can monitor temperature values, given randomly by the system and at the same time calculating the temperature difference between the last two temperature values.

Each time we have a critical sensing in a given cluster (i.e. temperature difference is greater than 3 degrees or we reach the maximum temperature), a sensing is sent to the cluster head (marked by a red line display from the cluster head itself to the base station).

Figure 4.5 below explains our simulation model in more detail. It shows the critical sensing in red sent from cluster head to the base station, and calculates the sensing number arrivals to the base station (4 sensings in this example)



Fire Detectors Simulator

Initial Temperature 20

Sensing Delay in Sec 1

UpperBound Temp. 26

Alarm Temp 25

OR Temp Diff. 3

LowerBound Temp 19

Figure 4.5: Fire Detector Critical Sensing

4.5 Experimental results

In order to determine an approximate value of the cluster head period "*T*" we have undertaken several sets of experiments.

4.5.1 Sensing Trace

The first set (Figure 4.6) below aimed at evaluating the number of sensings per second. In this test bed, we ran our simulation for a window period of 3500 sec. At *T*=500 sec. we have 10 sensings, at *T*=1000 sec. we read 16 sensings and so on until the end of the window period. So we decided to take a window of 500 sec. and meticulously analyze its sensing. What we found is presented in Figure 4.7 below.

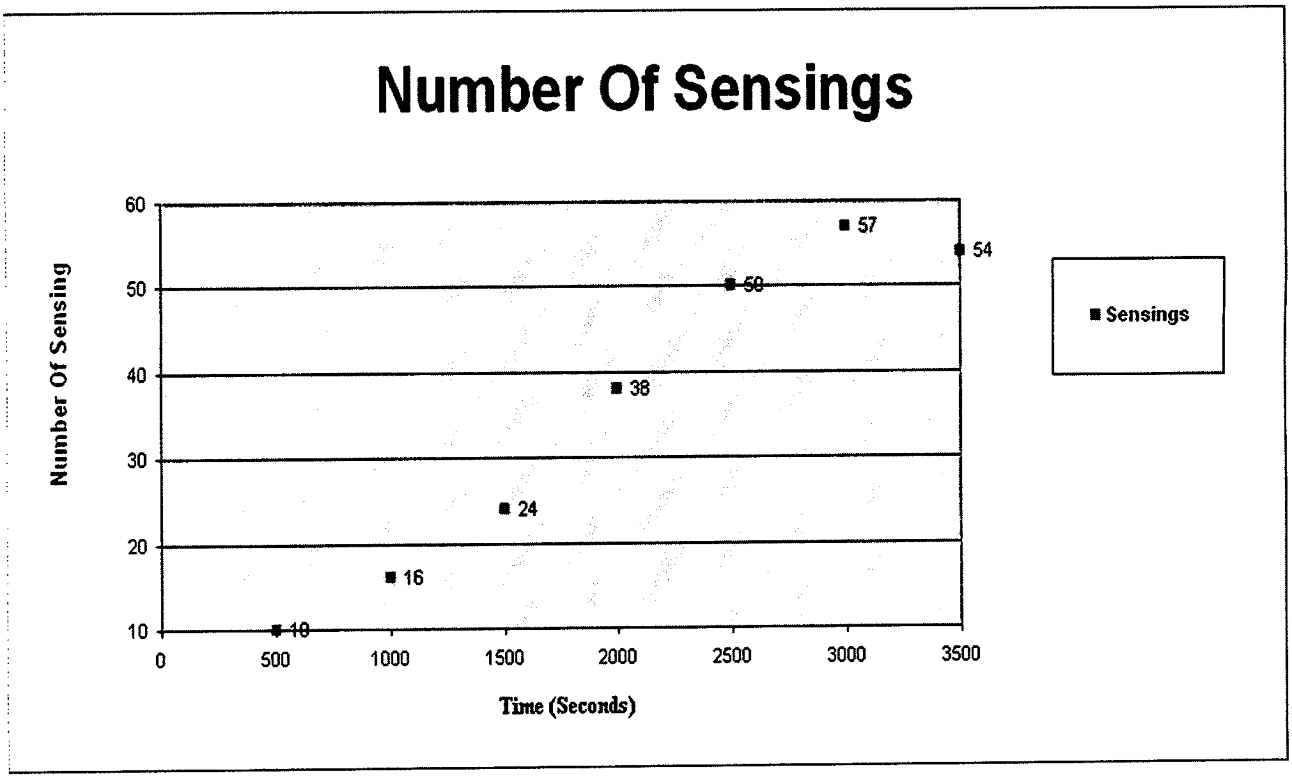


Figure 4.6: Number of Sensings per Second

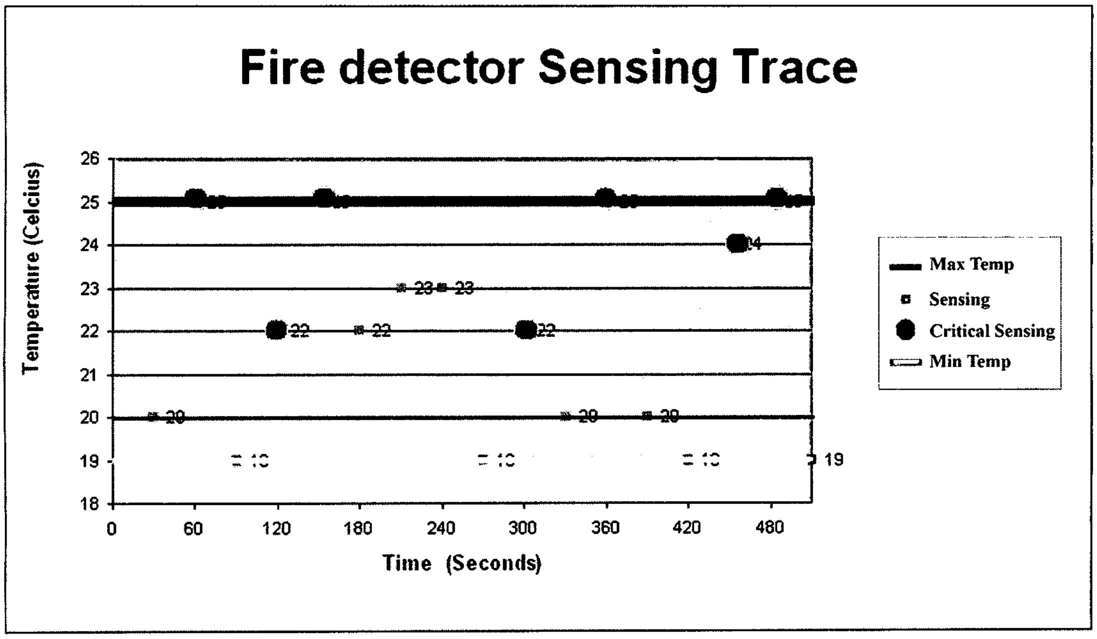


Figure 4.7 (a): Fire Detector Sensing Trace on 510 sec.

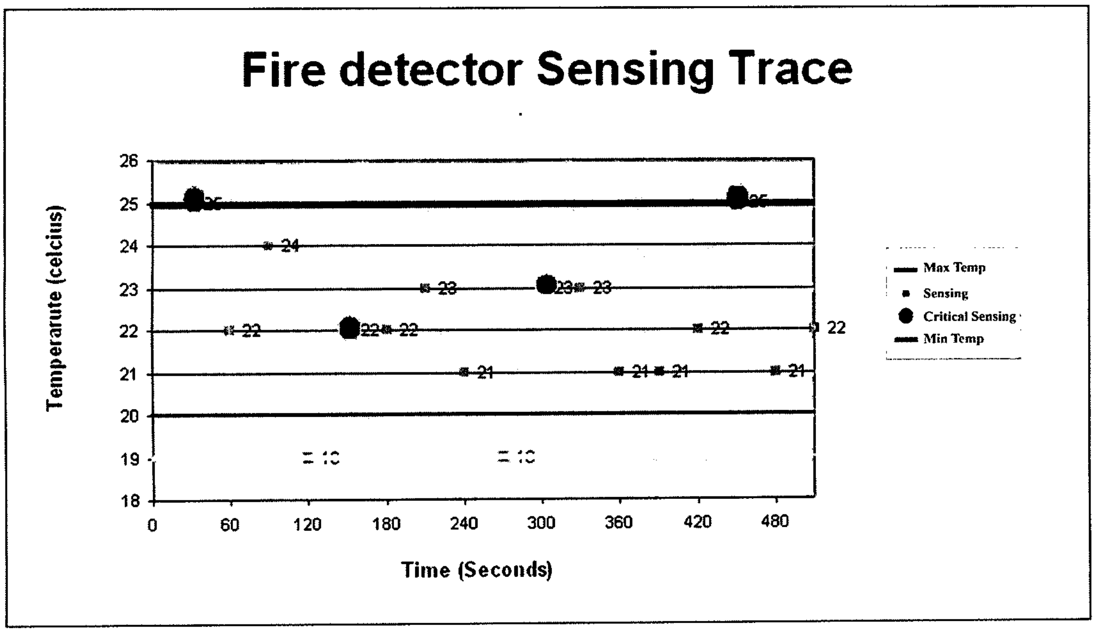


Figure 4.7 (b): Fire Detector Sensing Trace on 510 sec.

Figure 4.7 (a and b) were meant to identify the number of critical sensings. As we can see in these two figures the number of critical sensings varies from one simulation to another. Therefore, since we cannot at this stage fix the value of *T* more simulations are required. It is in fact important to check the impact of *T* variation on the whole system response time and energy consumption.

4.5.2 Response Time Analysis

By definition the response time is the average total time required by a given packet to traverse the whole WSN i.e the difference between its generation time at the source node and its arrival time to the sink.

In order to calculate the system response time, several assumptions were made:

- All generated packets have the same size and are equal to 20 Bytes.
- The wireless channel bit rate is equal to 20 Kbps (as recommended by the ZigBee standard [30]).
- CSMA/CA protocol is used between sensor nodes to avoid collision.
- The cluster head transmissions do not collide because an order is established among these nodes. Cluster head 1 starts its transmission first, then the second one and so on.

At multiple intervals of *T*, packets (depending on their availability) leave a given cluster head. Depending on the window of observation *T* (or the batching period *T*), the number of readings that fit in a given batch changes. As the simulation below shows, the response time is an increasing function of the cluster-head period *T*.

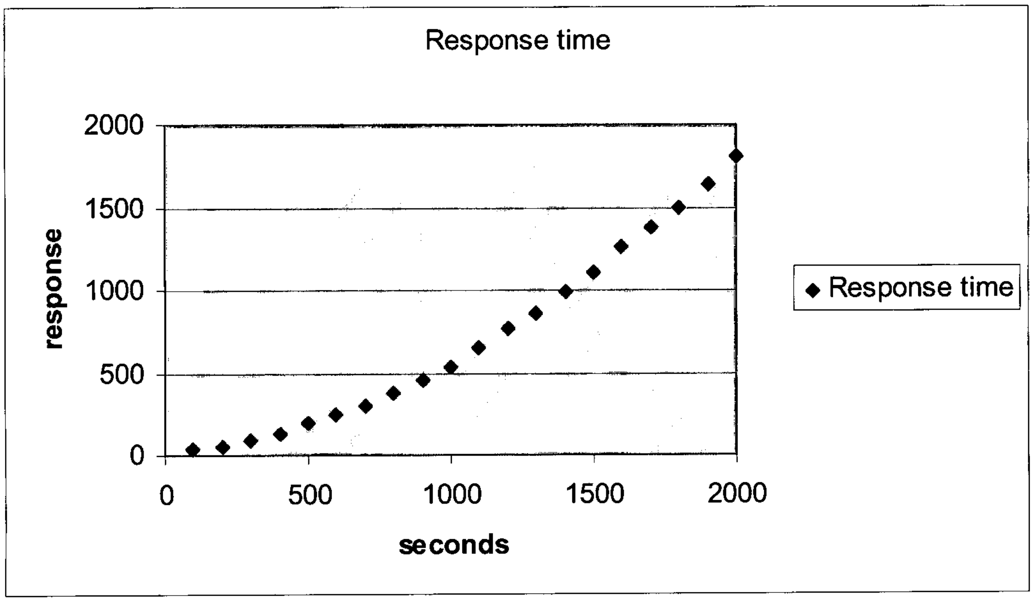


Figure 4.8: Impact of the Batching Period *T* on the Response Time

4.5.3 Energy Consumption Analysis

In what follows, we analyze the impact of the batching period *T* on the total energy consumption. We mean by total *energy consumption* the energy consumed by all the sensing nodes in the WSN during the whole simulation.

In this simulation we assume that on average a given sensor consumes 20 mjoules while transmitting its data.

As we can see in Figure 4.9 below, the total energy consumption is a decreasing function of the cluster-head period *T*. This is mainly due to the fact that the higher the value of *T*, the higher the number of concatenated sensing data at the cluster head and the smaller the energy needed for transmission (since there are fewer messages to transmit).

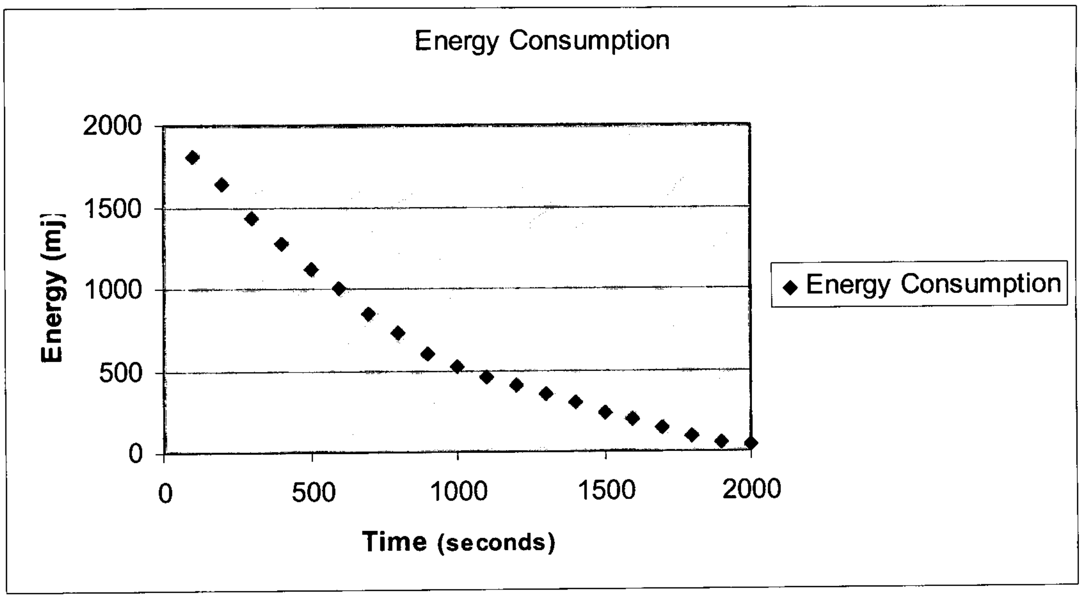


Figure 4.9: Total Energy Consumption Versus *T*

Therefore, it is evident that when we increase the value of *T* we minimize the total energy consumption in the system. However, we could also see from the previous paragraph, that an increase in the value of *T* will badly affect the system response time. So an optimum solution needs to be found i.e. an optimum value for *T* which minimizes both the delay and the energy consumption.

4.5.4 Optimum Solution

In order to find an optimum value for *T*, we calculated the normalized values of the response time and the energy consumption. This enabled us to put them on the same diagram. As we can see in Figure 4.10 below, an optimum value of *T* would be in the range of 800 sec.

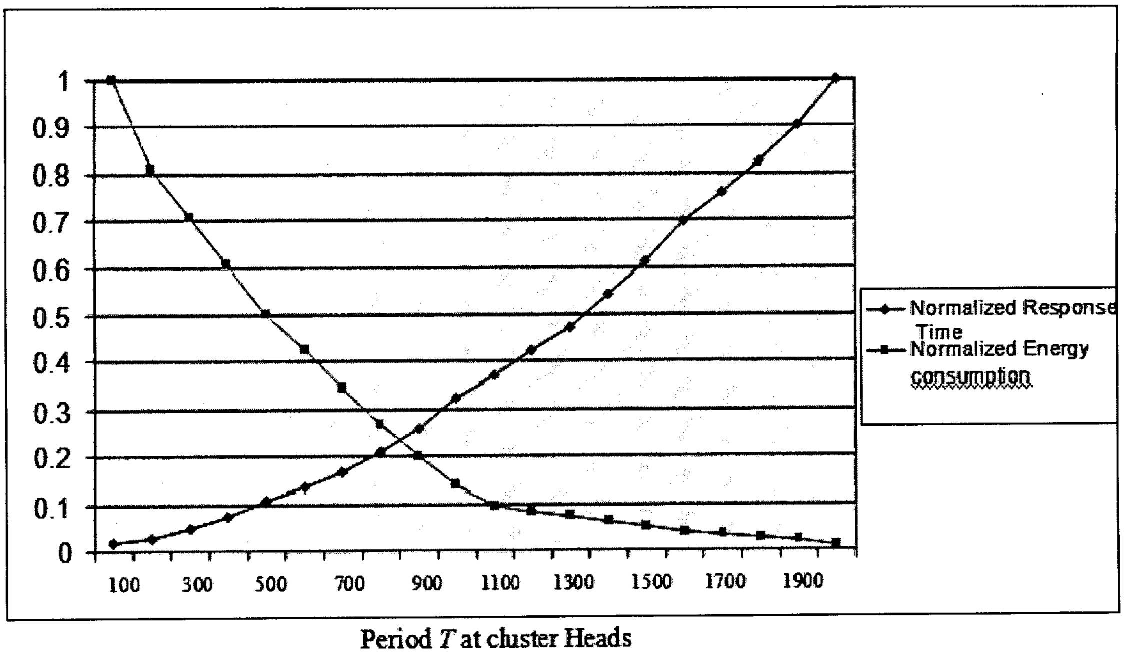


Figure 4.10: Normalized Values of the Delay and the Energy Consumption Versus *T*

Finally, we should note that the above found value for *T* is an optimum solution for the particular configuration that we have analyzed in this thesis. Finding a general solution could be rather difficult and would require a much larger simulation model in which the number of sensing nodes and cluster heads is variable.

Chapter 5

Conclusions

This chapter presents a summary of the efforts undertaken with regard to the proposition of the hybrid model for wireless sensor networks and provides the conclusions, the observations from this study and a discussion of possible future work.

5.1 Main Contributions

In this thesis we have developed a framework for improving WSNs' performance and management. This framework contributes to the field apart from technical bases for the evolution of this type of technology.

As defined in the text, some principles were considered in the conception of the model proposed, being simple adhering to network characteristics, including its dynamic behavior and being efficient in the use of scarce power resources. In spite of the rapid expansion of WSN research works, very little has concentrated on WSN fault management. Although this may not be a problem for small networks it certainly is for WSNs formed by hundreds or thousands of nodes which need the network and its elements to self-configure and adapt to their own state in the case of node or partial network failure.

In this thesis we proposed a hybrid WSN model that considers the application requirements and the sensor node operation constraints, to achieve low power consumption balanced against the required application responsiveness. In fact, the reduction of power consumption is crucial to increase the lifetime of low-power sensor networks.

We achieved low power consumption by exploiting the hybrid model scheme which is made of the concatenation of an event driven stage and a continuous stage. Cluster controllers were used as interfaces between the two stages of WSNs and played a major role in the communication model and the management scheme. Cluster controllers periodically (with a period *T*) transmitted their available data to the sink. We found that we can minimize the total energy consumption by changing the value of *T*. In fact, by making *T* large enough, the departing message from different cluster controllers will not collide with each other and their numbers will be smaller. Hence, an overall saving in power consumption.

However, having *T* very large will increase the system response time, so a compromise was required to optimize the system performance. A simulation model was built in this thesis and an optimum value of *T* was found.

We believe that the framework proposed in this thesis is a relevant contribution for the field, since very little work exists in the literature that studies in detail WSN management and proposes possible techniques to enhance it.

5.2 Proposed Future Work

This work can be extended in various ways. Some immediate extensions would be a generalization of the simulation model for the following purposes:

- Finding an optimum value of *T* for any number of sensors, clusters and cluster heads.
- Incorporating the impact of the management data packets on the system performance parameters (i.e. system response time and energy consumption).
- Investigating the impact of node failure (sensor and cluster-head nodes) on the system performance.

BIBLIOGRAPHY

[1] Alberto Cerpa, Jeremy Elson, Michael Hamilton, Jerry Zhao, Deborah Estrin, and Lewis Girod. Habitat monitoring: application driver for wireless communications technology. In *ACM SIGCOMM Workshop on Data Communication in Latin America and the Caribbean*, pages 20 - 41, San Jose, Costa Rica, April 2001.

[2] Andreas Savvides, Sung Park, and Mani B. Srivastava. On modeling networks of wireless microsensors. pages 318-319. In Joint international conference on Measure-ment and modeling of computer systems, Cambridge, MA, United States, June 2001.

[3] Antonio A. Loureiro, Jose Marcos S. Nogueira, Linnyer B. Ruiz, and Raquel A. Mini. Rede de sensores sem fio. pages 193-234. XXI Jornada de Atualizacao em Informatica do Congresso da Sociedade Brasileira de Computacao, julho 2002.

[4] Antonio A. Loureiro, Jose Marcos S. Nogueira, Linnyer B. Ruiz, Eduardo Nakamura, Carlos Mauricio Serodio, and Raquel Mini. Redes sensores sem fio. pages 179-226. Simposio Brasileiro de Redes de Computadores (SBRC), maio 2003.

[5] B. R. Badrinath, M. Srivastava, K. Mills, J. Scholtz, and K. Sollins. Special issue on smart spaces and environments. *IEEE Personal Communications*, October 2000.

[6] D. Estrin, R. Govindan, and J. Heidemann. Scalable coordination in sensor networks.Technical Report 99-692, University of Southern California, January 1999.

[7] D. Estrin, L. Girod, G. Pottie, and M. Srivastava. Instrumenting the world with wireless sensor networks. In International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2001), May 2001.

[8] Fabricio Silva, Thais R.M. Braga, Linnyer B. Ruiz, and Jose Marcos S. Nogueira.Tecnologia de nos sensores sem fio. Relatorio Tecnico RT.DCC/UFMG 006/2003,Departamento de Ciencia da Computacao, janeiro 2003.

[9] G. J. Pottie and W. J. Kaiser. Wireless sensor networks. *Communication ACM*, (5):51-58, May 2000.

[10] Ian Akyildiz, Weilian Su, Yogesh Sankarasubramaniam, and Erdal Cayirci. A survey on sensor networks. *IEEE Communication Magazine*, 40(8):102 -114, August 2002

[11] IEEE Communications Magazine – August 2002

[12] International Telecommunication Union (ITU). *ITU-T M.3010 - Principles for aTelecommunications management network*, May 1996.

[13] Jeremy Elson and Deborah Estrin. Random, ephemeral transaction identifiers in dynamic sensor networks. 21st International Conference on Distributed Computing Systems (ICDCS-21), Phoenix, Arizona, USA, April 2001

[14] John S. Heidemann, Fabio Silva, Chalermek Intanagonwiwat, Ramesh. Govindan,Deborah Estrin, and Deepak Ganesan. Building efficient wireless sensor networks with low-level naming. In *Symposium on Operating Systems Principles*, pages 146-159, 2001

[15] JPL Sensor Webs. Available in http://sensorwebs.jpl.nasa.gov/, February 2002.

[16] K. Sohrabi, J. Gao, V. Ailawadhi, and G.J. Pottie. Protocols for self-organization of a wireless sensor network. *IEEE Personal Communications*, 7(5):16-27, October 2000.

[17] L. Nirupama, B. Deborah, and E. Deborah. Scalable coordination for wireless sensor networks: Self-configuring localization systems. International Symposium on communication Theory and Applications (ISCTA), July 2001.

[18] Linnyer B. Ruiz, Jose Marcos S. Nogueira, and A. A. Loureiro. *Handbook of Sensor Networks: Compact Wireless and Wired Sensing Systems*, volume 1, chapter Sensor Network Management. Edited by Mohammad Ilyas and Imad Mahgoub, CRCPress, June 2004.

[19] Linnyer B. Ruiz, Federal University of Minas Gerais and Pontifical Catholic University of Parana, Jose Marcos S. Nogueira and Antonio A. Loureiro, Federal University of Minas Gerais . MANNA: A Management Architecture for Wireless Sensor Networks, Dec 2003.

[20] Loren Schwiebert, Sandeep K. S. Gupta, and Jennifer Weinmann. Research challenges in wireless networks of biomedical sensors. In *Mobile Computing and Networking*, pages 151-165, 2001.

[21] Madhavi W. Subbarao. Ad hoc networking critical features and performance metrics. Technical report, Wireless Communications Technology Group, NIST, September 1999.

[22] Marcos A. Vieira, Luiz Filipe Vieira, Linnyer B. Ruiz, , Antonio A. Loureiro, Antonio O. Fernandes, and Jose Marcos S. Nogueira. Scheduling nodes in wireless sensor network: A voronoi approach. *IEEE LCN - Local Computer Network*, pages 423 - 429, October 2003

[23] Raquel A. F. Mini, Badri Nath, and Antonio A. F. Loureiro. A probabilistic approach to predict the energy consumption in wireless sensor networks.

[24] Robert Hills. Sensing for danger. Technical Report UCRL-52000-01-7/8, Lawrence Livermore National Laboratory, http://www.llnl.gov/str/JulAug01/Hills.html, August 2001

[25] S. Park, A. Savvides, and M. B. Srivastava. Sensorsim: a simulation framework for sensor networks. In *Proceedings of the 3rd ACM international workshop on Modeling, analysis and simulation of wireless and mobile systems*, pages 104-111, Boston, MA, United States, 2000

[26] Smart Dust - autonomous sensing and communication in a cubic millimeter. Available in http://robotics.eecs.berkeley.edu/epister/smartdust, February 2002.

[27] Shashank Mehrotra. Distributed algorithms for tasking large sensor network. *Thesis submitted to the Faculty of Virginia Polytechnic Institute and State Univesity*, July 2001.

[28] Stephanie Lindsey, Cauligi Raghavendra, and Krishna Sivalingam. Data gathering in sensor networks using the energy delay metric. In *International Workshop on Parallel and Distributed Computing: Issues in Wireless Networks and Mobile Computing*, April 2001

[29] Wireless Integrated Network Sensors (WINS). Available in http://www.janet.ucla.edu/wins/March 2002

[30] ZigBee Alliance – Working with Wireless Control ; http://www.zigbee.org/