ENERGY-SAVING HYBRID DETECTION SCHEME WITH PATH LENGTH AWARE FOR WIRELESS SENSOR NETWORK

BY

MOHAMMAD NABEEL AL-SALLOUT

A Thesis Presented to the
DEANSHIP OF GRADUATE STUDIES

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

In

COMPUTER ENGINEERING

MAY 2012

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS
DHAHRAN 31261, SAUDI ARABIA

DEANSHIP OF GRADUATE STUDIES

This thesis, written by **MOHAMMAD NABEEL AL-SALLOUT** under the direction of his thesis adviser and approved by his thesis committee, has been presented to and accepted by the Dean of Graduate Studies, in partial fulfillment of the requirements for the degree of **MASTER OF SCIENCE IN COMPUTER ENGINEERING**.

Thesis Committee

Dr. Tarek Sheltami (Advisor)

Dr. Ashraf Sharif (Member)

Mr. Waseem Raad (Member)

Dr. Bassam Al-Madani
Department Chairman

Dr. Salam A. Zummo
Dean of Graduate Studies

21/5/12
Date

DEDIACATION

To my wife and my son Nabeel for their patience.

To my father Nabeel and my mother Zainab for their support.

To all my brothers and sisters.

To all my respectful Teachers.

ACKNOWLEDGMENTS

All praise and glory are due to Allah, to Him belongs the sovereignty of the heavens and the earth. May His blessings and mercies be upon the noblest of mankind, Muhammad (S.A.W.), his household, his companions and the generality of the true believers to the last day. I am grateful to Allah for all His favours on me since my birth, these blessings are indeed innumerable, and the greatest of His bounties on me is being a Muslim.

wish to express my gratitude to my parents, siblings, uncles and aunties for their support and kind words.

My warmest appreciation goes to my thesis advisor, Professor Tarel Sheltami for his ntellectual guidance, support and readiness to assist during this research and even before this research. Also, I appreciate the helpful suggestions and contributions of other thesis committee members: Professor Ashraf Sharif and Dr. Wassem Raad.

TABLE OF CONTENTS

DEDIACATION iii
ACKNOWLEDGMENTS..iv
Table of Contents v
List of Tables..vii
List of Figures viii
Thesis Abstract (English) x
Thesis Abstract (Arabic) xii
Chapter 1 Introduction..1
1.1 Background..1
1.2 Motivation..2
1.3 Thesis Objective..4
1.4 Thesis Structure..4
Chapter 2 Wireless Sensor Networks..6
2.1 Overview of WSN..6
2.2 Wireless Sensor Nodes..7
2.3 WSN Applications 9
2.3.1 Medical Monitoring 9
2.3.2 Industrial Automation 9
2.3.3 Civil Structure Monitoring..10
2.4 Communication Technology..11
2.4.1 Physical Layer 11
2.4.2 MAC Layer..13
2.4.3 Network Topology..15
2.5 Standards and specifications 17
2.5.1 IEEE 802.15.4 (WSN) 18
2.6 Operating Systems and Prototypes..23
2.6.1 Available OS..26

2.7 WSN Simulators 34
2.7.1 Specific Framework Simulator 34
2.7.2 General Simulation Tool 35
2.7.3 Omnet++ (commercial version OMNEST) 37
Chapter 3 Related Work 41
3.1 Bayesian Decision Theory 42
3.1.1 Binomial Distribution 42
3.1.2 Bayes' Rule 43
3.2 Centralized Detection Schemes 44
3.3 Distributed Detection Scheme 46
3.4 Hybrid Detection Scheme 48
Chapter 4 Research Methodology 51
4.1 Problem Statement 51
4.2 System Model (TELOS) 52
4.3 Power Calculation: 54
4.3.1 Processing Energy: 55
4.3.2 Transmission Energy: 55
4.4 Simulation Network Design 56
Chapter 5 Enhanced Hybrid Detection Scheme 57
5.1 Network Deployment 59
5.2 Nodes Configuration 60
5.3 N_0, N_1 Selection 62
5.3.1 All N_0, N_1 combinations: 62
5.3.2 Valid N_0 and N_1 combinations: 63
5.3.3 Map (N_0, N_1) combinations to available zones: 64
5.4 Final Decision: 67
Chapter 6 SIMULATION Results 69
6.1 Enhanced Hybrid Detection Scheme 69
6.2 Power Consumption: 70
6.3 Accuracy behavior: 77
Chapter 7 Conclusion 83
Chapter 8 References 85
Vita 92

LIST OF TABLES

TABLE 1 Industrial, Scientific and Medical (ISM) Bands 12
TABLE 2 WSN MAC Protocols 14
TABLE 3 Classification of Routing Protocols in WSNs 16
TABLE 4 Wireless Communication Protocols 18
TABLE 5 Transmission Parameters of IEEE 802.15.4 20
TABLE 6 WSN Available Operating Systems 26
TABLE 7 Operating System and Supported Platforms 34
TABLE 8 Telos Typical Operation Conditions [50] 53
TABLE 9 CP Values for all (N_0,N_1) Combinations 63
TABLE 10 (N_0,N_1) Valid Combination List 64
TABLE 11 Zones to (N_0,N_1) Cominations Mapping 66
TABLE 12 Zones Usage Percantage 69
TABLE 13 Centrlized Scheme Power Consumption 71
TABLE 14 Distributed Scheme Power Consumption 72
TABLE 15 Hybrid Scheme Power Consumption 73
TABLE 16 All (N_0,N_1) Combinations 74
TABLE 17 Enhanced Hybrid Scheme Power Consumption 74
TABLE 18 Power Consumption of all Schemes 75
TABLE 19 Centrlized Scheme Propability of Error 78
TABLE 20 Distributed Scheme Propability of Error 79
TABLE 21 Hybrid Scheme Propability of Error 80
TABLE 22 Enhaced Scheme Propability of Error 81
TABLE 23 Propability of Error for all Schemes 82

LIST OF FIGURES

Fig. 1. Berkeley Mote....3
Fig. 2. Typical Multi-hop Wireless Sensor Network Architecture [7]....6
Fig. 3. Main Sensor Node Function [2]....8
Fig. 4. Sensor Node Architecture [8]....8
Fig. 5. Industrial Application of Wireless Sensors....10
Fig. 6. Ben Franklin Bridge....10
Fig. 7. Bridge Strain Data....11
Fig. 8. Categories of MAC Protocols....13
Fig. 9. Main Sensor Network Topologies....15
Fig. 10. Categories of routing protocols [20] [2]....16
Fig. 11. Available Wireless Standards....17
Fig. 12. Structure of IEEE 802.15.4 Protocol Stack....19
Fig. 13. ZigBee Protocol Stack....21
Fig. 14. Elements of a Typical WirelessHART Installation....22
Fig. 15. ISA100.11a Stack....23
Fig. 16. Middleware Implementation....24
Fig. 17. WSN Operating System Features....25
Fig. 18. Simplified TinyOS Architecture....27
Fig. 19. Screenshot of the VNC Server Running on the Atmel AVR Port of Contiki....29
Fig. 20. Simplified Contiki Kernel Architecture....29
Fig. 21. Simplified MANTIS OS Kernel Architecture....31
Fig. 22. SOS Architecture....32
Fig. 23. Layered Architecture of the .NET Micro Framework....33
Fig. 24. PAWiS Architecture....39

Fig. 25. Data Processing Tool 40
Fig. 26. Conditional Probability Example 43
Fig. 27. TelosB by the University of California 52
Fig. 28. Power Consumption of "Telos" 53
Fig. 29. An Example of Sensor Deployment [52] 56
Fig. 30. Node Behaviour Depends on N_0 and N_1 58
Fig. 31. Sample Sensor Mesh Network 59
Fig. 32. Mapping (N_0,N_1) Combinations to Available Path Length 61
Fig. 33. Network Zones Classification 65
Fig. 34. Nodes Average Path Length 70
Fig. 35. Centrlized Scheme Power Consumption 71
Fig. 36. Distributed Scheme Power Consumption 72
Fig. 37. Hybrid Scheme Power Consumption 73
Fig. 38. Enhanced Hybrid Scheme Power Consumption 75
Fig. 39. Power Consumption of all Schemes 76
Fig. 40. Enhanced vs. Hybrid Power Consumption 76
Fig. 41. Comparison of Three Schemes in Detection Accuracy [46] 77
Fig. 42. Centrlized Scheme Propability of Error 78
Fig. 43. Distributed Scheme Propability of Error 79
Fig. 44. Hybrid Scheme Propability of Error 80
Fig. 45. Enhaced Scheme Propability of Error 81
Fig. 46. Propability of Error for all Schemes 82

THESIS ABSTRACT

NAME: Mohammad Nabeel Mousa Al-Sallout

TITLE OF STUDY: Energy-Saving Hybrid Detection Scheme With Path Length Aware For Wireless Sensor Network

MAJOR FIELD: Computer Engineering

DATE OF DEGREE: November 2011

Wireless sensor networks are used today in a wide range of applications, all of which employ a large number of sensors. In large scale sensor networks, sensor nodes are often not easily accessible. Because of this, the energy consumption of wireless networks is an important matter and a popular topic of research.

A sensor node consumes energy while collecting, processing, transmitting and receiving data. Each of these processes could be the focus of research, so there are many investigations into these subjects, centering on ways of reducing energy consumption and extending the lifetimes of networks.

In this thesis I study data processing schemes that define the distribution of decision making, which affects system accuracy and energy consumption. The two typical detection schemes are the centralized and distributed schemes. In a centralized scheme, nodes collect samples from the environment and send them to a "fusion center", where the samples are used to arrive at a final decision. This scheme provides optimal decision *accuracy; however, it consumes considerable energy. In contrast, distributed schemes allow nodes to make local 1-bit decisions which are sent to the fusion center to make the final decision. In a hybrid scheme the network specifies the level of accuracy required for the whole system. This can be achieved by manipulating the scheme to work sometimes as centralized other times as distributed. I propose an energy-saving hybrid scheme which focuses on optimizing transmission energy, since most of the energy consumed is in the transmission process. In the proposed scheme each node will vary between centralized and distributed according to its location and path length. Nodes with longer path lengths are classified as acting more as distributed than nodes with shorter path lengths.*

Keywords: *WSN, sensor node, centralized, distributed, data processing, energy consumption, fusion center.*

ملخص الرسالة

الإسم: محمد نبيل موسى السلوت

عنوان الرسالة: نظام الكشف الهجين الموفر للطاقة عبر ادراك طول المسار لشبكة الإستشعار اللاسلكية

التخصص: هندسة الحاسب الآلي

تاريخ التخرج: مايو 2012

تستخدم شبكات الاستشعار اللاسلكية اليوم في مجموعة واسعة من التطبيقات، والتي تقوم بتوظيف عددا كبيرا من أجهزة الاستشعار. غالباً في شبكات الاستشعار التي تغطي مساحات واسعة، يكون الوصول الى عقد الاستشعار المنتشرة ليس بالأمر السهل. لهذا السبب، فإن استهلاك الطاقة في الشبكات اللاسلكية هي مسألة هامة وموضوعاً مطروحاً للبحث.

تستهلك عقدة استشعار الطاقة خلال جمع وتجهيز وارسال واستقبال البيانات. ويمكن لأي من هذه العمليات أن تكون محوراً للبحث، لذلك فإن هنالك العديد من البحوث التي تطرقت لهذه المواضيع، وتركزت حول سبل الحد من استهلاك الطاقة وإطالة عمر شبكات الإستشعار اللاسلكية.

في هذه الرسالة قمت بدراسة انظمة معالجة البيانات التي تحدد مكان صنع القرار في الشركة، والذي يؤثر على دقة استشعار الشبكة ومقدار استهلاكها للطاقة. نطرح في البداية نظامين رئيسيين لمعالجة البيانات هما النظام المركزي والنظام الموزع. في النظام المركزي، تقوم عقد الإستشعار بجمع عينات من القراءات للبيئة المحيطة وإرسالها إلى "مركز المعالجة الرئيسي"، حيث يتم استخدام العينات للتوصل إلى قرار نهائي. وهذا النظام يعتبر الأكثر دقة في اتخاذ القرار، وفي المقابل فأنه يستهلك قدرا كبيرا. على الجانب الآخر،فإن النظام الموزع يسمح لعقد الإستشعار لإتخاذ قرار محلي وارسال ذلك القرار على شكل 1 بت إلى مركز المعالجة الرئيسي لاتخاذ القرار النهائي. في النظام الهجين تقوم الشبكة باختيار مستوى من الدقة المطلوبة للنظام بأكمله. ويمكن تحقيق هذا الإختيار عن طريق التحكم في النظام ليقوم بالعمل كنظام مركزي احياناً وكنظام موزع في أحيان أخرى. في هذه الرسالة قمت بابتكار نظام هجين جديد لتوفير الطاقة والتي تركز على ترشيد استهلاك الطاقة المستهلكة في ارسال البيانات، لأن معظم الطاقة المستخدم في الشبكة تستهلك في عمليات ارسال البيانات. في النظام المستحدث سوف يختلف نمط اتخاذ القرار لكل عقدة حسب توزيعها ووفقا لموقعها الجغرافي وطول مسارها. لذلك فإن العقد ذات المسار الأطول تتخذ النمط الموزع بشكل اكبر من العقد ذات المسار الأقصر.

CHAPTER 1

INTRODUCTION

In this chapter, background information about wireless sensor networks (WSNs) is introduced in Section 1.1; in Section 1.2 the problem is stated and thesis motivation is discussed; in Section 1.3 the objective of the thesis is specified; and in Section 1.4 the overall report structure of the thesis is presented.

1.1 Background

The function of sensors is to convey physical phenomena to the digital world by capturing and revealing real-world behaviors and converting these phenomena into a form that can be processed, stored, and acted upon.

With sensor capabilities a tremendous societal benefit is achieved when sensors are integrated into available devices, machines, and environments. They can help to avoid infrastructure failure disasters, protect precious natural resources, enhance security and enable new "smart" applications such as context-aware systems and smart home technologies.

Wireless sensor networks are based on numerous advanced technologies such as very large scale integration (VLSI), micro-electromechanical systems (MEMS) and wireless communications. The development of these technologies is contributing to a wider application of WSNs. For example, with the enhancement of MEMS technology, sensors are becoming smaller, and developments in semiconductor technologies are producing smaller microprocessors with higher processing capacities. This improvement of computing and sensing technologies is enabling the development of flexible WSNs, which can be widely applied [1] [2].

1.2 Motivation

Monitoring environmental changes and detecting specified events is the main function of sensor networks. This function is achieved through four basic components of a sensor network [3]:

1. distributed or localized sensors
2. an interconnecting network – most often wireless based
3. a central point of information clustering
4. a set of computing resources at the central point or network core to handle data collecting, event trending, status querying, and data mining.

WSNs use centralized fusion centers (sinks), which work as cluster gateways, and many distributed sensors (motes) [4]. These sensors sense and send observations to the centralized unit. The centralized unit decides if an event is initiated or not. Fig. 1 shows an example of a sensor node.

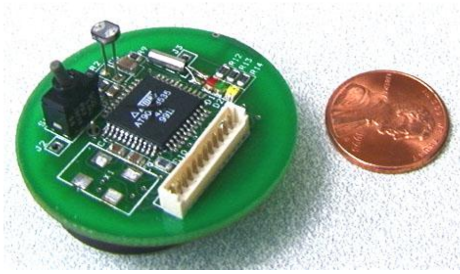

Fig. 1. Berkeley Mote

Most of the power consumed in a network is used in processing, transmitting and sensing. Until now limited power resources for sensors has been the main constraint in WSNs. It is very important to reduce sensor power consumption while maintaining acceptable detection accuracy according to application requirements. Many researchers have focused on the above three processes [5], attempting to enhance the power consumption efficiency of the sensors for each of them. Some schemes enhance the operating system and reduce the required processing cycles; other schemes optimize the RF part including collision space and noise filtering. This thesis focuses on schemes which study decision processing and transmitting where those schemes define how to collect observations (sampling rate), where to process them (locally or centralized) and the data to be sent from nodes to the fusion center, which will affect the degree of loss of data and accuracy [6].

1.3 Thesis Objective

The main objective of this thesis is to produce an optimum controlling scheme that extends network/sensor lifetime by reducing power consumption and minimizing the loss of network efficiency and accuracy. Our proposed scheme controls and defines the processing and transmission functionality of the wireless sensor network. This scheme balances the reduction of data transmission and detection processing by distributing these two activities between nodes and the central unit (sink).

1.4 Thesis Structure

The thesis is composed of the following chapters:

Chapter 2: This chapter presents an overview of WSNs and covers the technology used. It includes functionality, hardware, networking and software. This chapter also presents some real world applications of WSNs.

Chapter 3: In this chapter previous related works are introduced and the main features and differences within those works are compared.

Chapter 4: Here the problem is stated, and the target and research methodology is presented.

Chapter 5: The present work is compared with older schemes.

In **Chapter 6** our simulation system is presented, and the system model is defined. Then our results are displayed, with graphs showing the efficiency of our scheme compared with available techniques.

Finally in the **Conclusion** our observations and achievements are summarized.

CHAPTER 2

WIRELESS SENSOR NETWORKS

2.1 Overview of WSN

A wireless sensor network (WSN) consists of a large number (from a few hundred to thousands) of dispersed independent sensors which have the functionality of monitoring physical conditions such as wind, temperature, sound, vibration or pressure. Fig. 2 shows a basic example of a WSN.

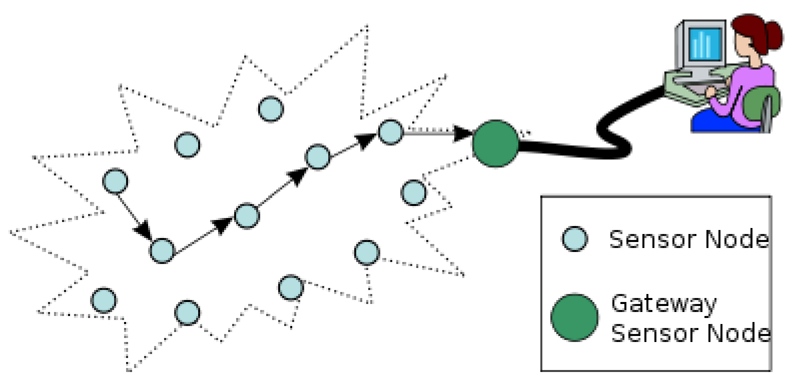

Fig. 2. Typical Multi-hop Wireless Sensor Network Architecture [7]

In the early days of this technology, wireless sensor networks were developed for military applications such as battlefield surveillance. But later, when sensor nodes became smaller and cheaper, WSNs began to be used for a wide range of applications. In the industrial area, they are used to monitor and control the industrial processes. In the geophysical and environmental area earthquakes and volcanoes could be monitored. Civilian usage included smart home automation and traffic control. Nowadays developers are interested in personal area networks (PANs) such as healthcare applications.

2.2 Wireless Sensor Nodes

A sensor node is a device that contains at least one sensor and may also include actuators. This node must have processing and networking capabilities to process data and use wireless access.

Sensors can measure quantity and convert the measurement into a signal. The measurement may be electrical, mechanical, chemical, acoustic, or of any other physical type. Fig. 3 illustrates that a sensor acts like an interface between its environment and a digital system such as a computer. Typically, sensors are small, cheap, and low power-consumption devices requiring a limited amount of information transfer.

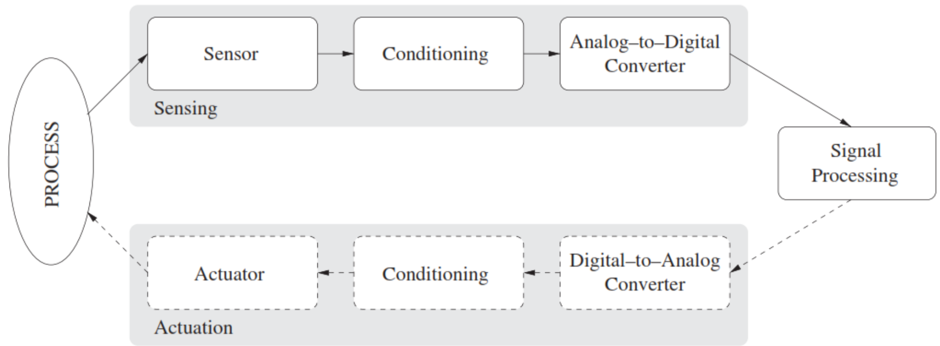

Fig. 3. Main Sensor Node Function [2]

A sensor node is an appropriate application for a "system on a chip" (SoC) that has communication, computation, sensing and storage capabilities [8] [9]. A wide variety of platforms have been developed in recent years, including Mica2 [10], Cricket [11], MicaZ, Iris, Telos [10], SunSPOT [12], and Imote2.

Basically, each node is composed of a micro-controller, a power source, a radio frequency (RF) transceiver, external memory, and sensors, as shown in Fig. 4.

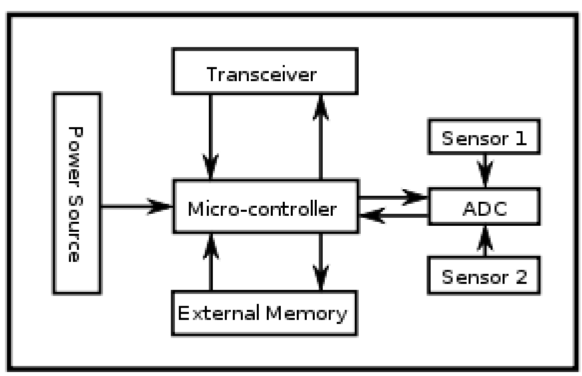

Fig. 4. Sensor Node Architecture [8]

In summary, a wireless sensor node - known as a 'mote' – can be defined as an end node in a wireless sensor network, which has the capability to gather and process information and communicate with other connected nodes in the network.

2.3 WSN Applications

Wireless sensor networks find applications in many different areas.

2.3.1 Medical Monitoring

Medical applications are designed to protect patients who are in danger of sudden health emergencies. Sensors can be fixed to their clothes, and warning signs can be detected and sent through a network. On the basis of such detection proper actions can be taken by the family, a doctor or the person himself.

2.3.2 Industrial Automation

In addition to the high cost of using wired sensors in the automotive industry, they are also difficult to use for moving parts such as wheels. The use of wireless sensors allows for the rapid installation of sensing equipment and permits access to locations that would not be practical if cables were attached. It also makes repair operations easier and cheaper [5]. Fig. 5 illustrates the use of sensors in automobile manufacture.

Fig. 5. Industrial Application of Wireless Sensors

2.3.3 Civil Structure Monitoring

One of the most popular applications of today's sensor networks is in the monitoring of the structural health of large civil structures, such as the Ben Franklin Bridge in Fig. 6 [5].


Fig. 6. Ben Franklin Bridge

Here sensors operate in a low-power sampling mode in which they check for the presence of a train by sampling strain sensors at a low sampling rate (as seen in Fig. 7) and then reporting this data.

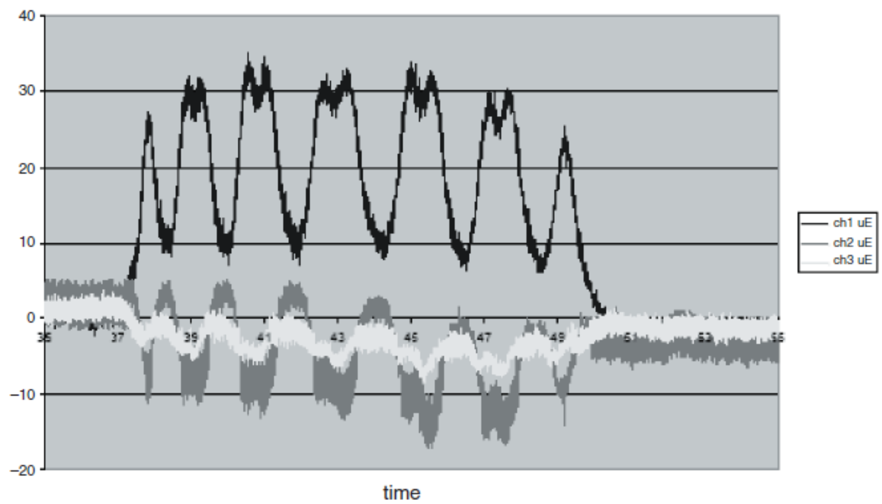

Fig. 7. Bridge Strain Data

2.4 Communication Technology

In this section an overview of general WSN technology including communication technology and available WSN platforms is presented.

2.4.1 Physical Layer

Now a day a wireless links is the most commonly medium used to communicating nodes in sensor networks. These links can be radio or infrared or optical media. If we want to decide which medium to use we should put in mind that, transmission medium must be available worldwide. So for radio medium we can select from the license-free bands listed in industrial, scientific and medical (ISM) bands which is listed in TABLE [13] [14] [2].

TABLE 1
Industrial, Scientific and Medical (ISM) Bands

Spectrum	Center frequency	Availability
6.765–6.795 MHz	6.780 MHz	Subject to local regulations
13.55–13.56 MHz	13.56 MHz	
26.95–27.28 MHz	27.12 MHz	
40.66–40.70 MHz	40.68 MHz	
433.0–434.7 MHz	433.9 MHz	Europe, Africa, the Middle East west of the Arabian Gulf
902.0–928.0 MHz	915.0 MHz	The Americas, Greenland and some of eastern Pacific Islands
2.400–2.500 GHz	2.450 GHz	(used by IEEE 802.11b, Bluetooth, IEEE 802.15.4 WPAN)
5.720–5.875 GHz	5.800 GHz	
24.00–24.25 GHz	24.12 GHz	
61.00–61.50 GHz	61.25 GHz	Subject to local regulations
122.0–123.0 GHz	122.5 GHz	Subject to local regulations
244.0–246.0 GHz	245.0 GHz	Subject to local regulations

In wireless sensor networks, power efficiency has a greater concern than bandwidth, as the sensors generate small-volume data.

According to reference [15], in 2.4-GHz CMOS transceivers, for a radiated power of 0 dBm, the transmitter actually uses 32 mW, whereas the receiver uses even more, 38 mW, which means 35 mW on average. For Mica motes, 21 mW are consumed in transmitting mode and 15 mW in receive mode [14]. On the other hand, with ESB sensor nodes at 4.5 volts the cost is 54 mW for transmission and 20 mW for reception.

2.4.2 MAC Layer

The main functions of the MAC protocols are to establish communication links between sensor nodes and to share the medium band fairly and efficiently. At the same time energy consumption, throughput and latency should be kept in mind. In Fig. 8 the main MAC protocol classifications are shown and categorized based on functionality.

Since energy and power resources are limited in WSNs, there are several factors in MAC layering which can easily result in the wasteful use of these resources [16], i.e. collision recovery, packet forwarding, overhead and listening mode.

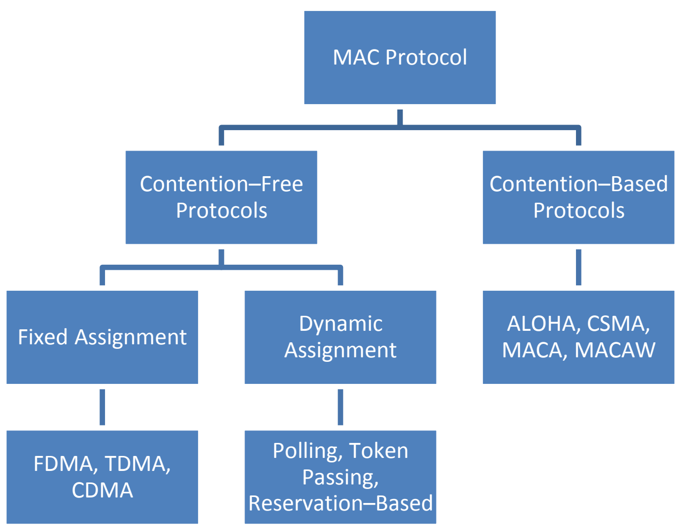

Fig. 8. Categories of MAC Protocols

Almost all MAC protocols used in WSN and in wireless networks in general use CSMA. TABLE 2 lists these protocols [2] [16]:

TABLE 2
WSN MAC Protocols

Protocol	Type	Description
IEEE 802.11	CSMA/CA , MACAW	General Standard for WLAN
IEEE 802.15.4	CSMA/CA	Designed for low power/rate devices
PAMAS	MACA	Power Aware Multi-Access Signaling
S-MAC [17]	CSMA/CA , MACAW	Modified IEEE 802.11 to save power.
T-MAC	CSMA/CA , MACAW	Enhanced S-MAC to save more power.
Optimized MAC	CSMA/CA , MACAW	Minimized S-MAC control packet overhead
TRAMA	TDMA / CSMA	Traffic Adaptive Medium Access
SMACS	TDMA/FDMA-CDMA	Self-Organizing Medium Access Control
WiseMAC	ALOHA	Solves the preamble latency problem.
B-MAC	ALOHA	Berkeley MAC, Based on Aloha
Energy Aware TDMA	TDMA	Uses cluster tree topology, time domain is controlled by the cluster's gateway.
D-MAC [16] [18] [19]	CSMA	Data Gathering MAC, designed and optimized for tree based data gathering.
Z-MAC	CSMA / TDMA	Zebra MAC, hybrid of TDMA and CSMA
CC-MAC		Spatial correlation-based collaborative MAC
MH-MAC	TDMA/CDMA	Mobility Adaptive Hybrid MAC

It is obvious from the above table that there are no specific standards for WSN, and protocol efficiency may differ from one application to other.

2.4.3 Network Topology

Since communication in WSNs is wireless, the same topology as general WLAN is applied, with consideration of the WSN's limitation of power. These topologies are mainly Star Topology, Mesh Topology and Cluster Tree topology (see Fig. 9).

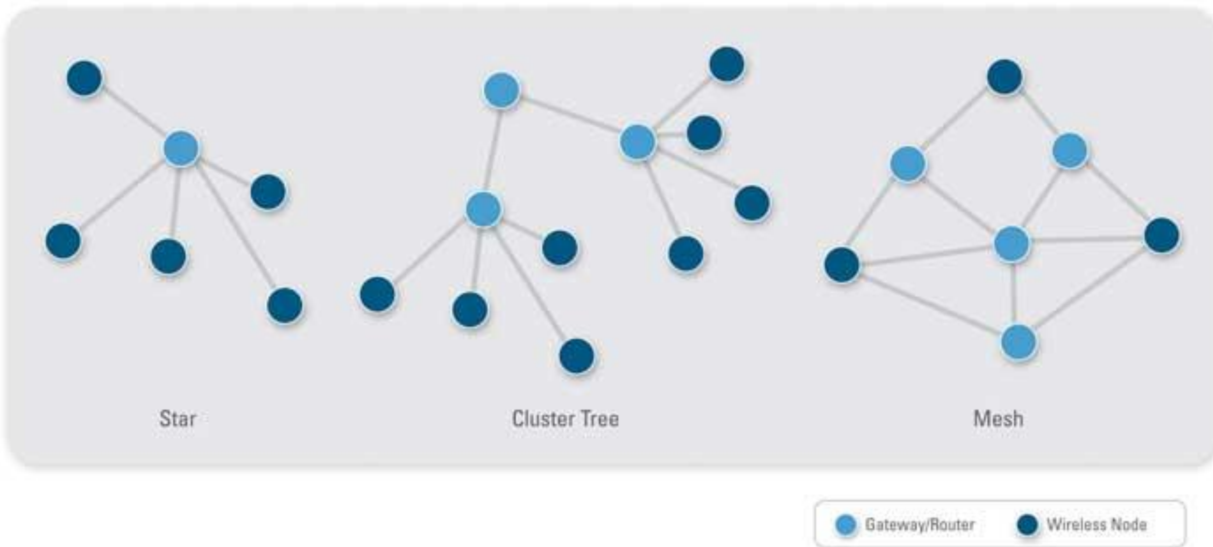

Fig. 9. Main Sensor Network Topologies

Although these three topologies can be used in WSNs, in reality some of them may be used more often than others, depending on the application and the challenges of the environment, which acts in this case like a WLAN. Fig. 10 shows the categorization of routing protocols, which is a serious matter for all wireless networks. Many protocols have been defined (see TABLE), and each of these protocols uses many matrices to design and compare routing protocols.

I use a simple shortest-path algorithm to simplify the problem statement in my simulation. It is nevertheless an efficient routing protocol, based on routes discovered at deployment.

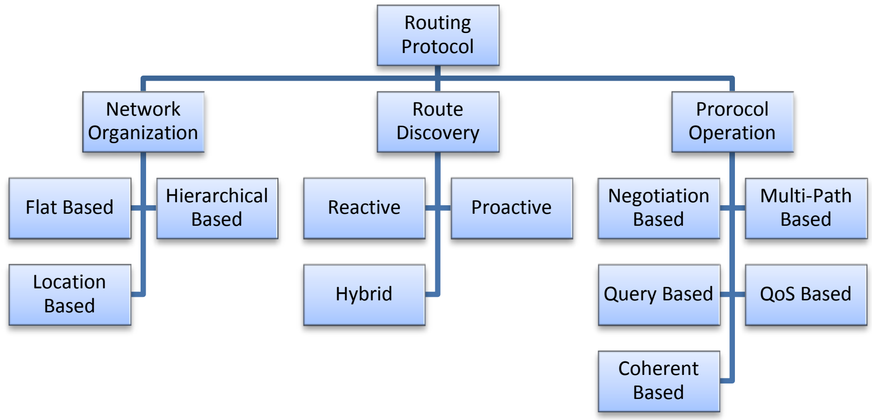

Fig. 10. Categories of routing protocols [20] [2]

TABLE 3
Classification of Routing Protocols in WSNs

Routing Protocol	
SPIN	Flat topology, data-centric, query-based, negotiation-based
Directed Diffusion	Flat topology, data-centric, query-based, negotiation-based
Rumor routing	Flat topology, data-centric, query-based
GBR	Flat topology, data-centric, query-based
DSDV	Flat topology with proactive route discovery
OLSR	Flat topology with proactive route discovery
AODV	Flat topology with reactive route discovery
DSR	Flat topology with reactive route discovery
LANMAR	Hierarchical with proactive route discovery
LEACH	Hierarchical, support of MAC layer
PEGASIS	Hierarchical
Safari	Hierarchical, hybrid route discovery (reactive near, proactive)
GPSR	Location-based, unicast
GAF	Location-based, unicast
SPBM	Location-based, multicast
GEAR	Location-based, geocast
GFPG	Location-based, geocast
SAR	Flat topology with QoS (real-time, reliability), multipath
SPEED	Location-based with QoS (real-time)
MMSPEED	Location-based with QoS (real-time, reliability)

2.5 Standards and specifications

All available standards are multi-layered. They usually cover physical and MAC layers, however; in some commercial standards higher layers are covered, which is why I list those standards with an explanation of each network's layers in the standard.

Currently there are many standards that can be followed in any wireless sensor network. Some of them are shown in Fig. 11. These standards already apply to many test-beds and products.

To compare standards, some factors will be used which vary from one application to another, i.e. coverage, bandwidth, data rate and power consumption [21].

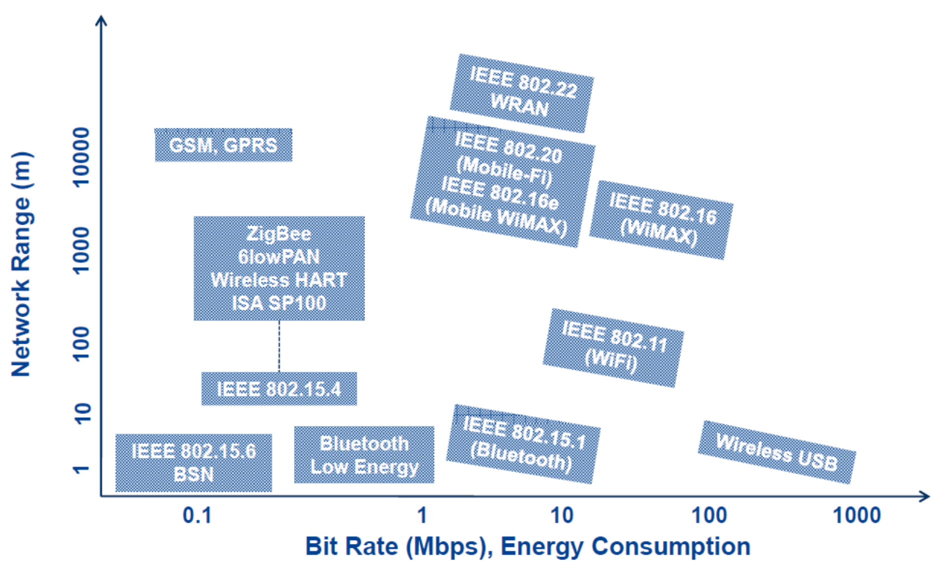

Fig. 11. Available Wireless Standards

Existing wireless communication protocols are shown in TABLE , including IEEE 802.11(WiFi), IEEE 802.15.1 (Bluetooth), IEEE 802.15.4 (ZigBee, 6LoWPAN, WirelessHART, ISA-100.11a), and IEEE 802.16 (WiMAX).

TABLE 4
Wireless Communication Protocols

Standards	IEEE 802.15.4 (Low Power-Wireless Personal Area Network) (LP-WPAN)				IEEE 802.15.1 (WPAN)	IEEE 802.11 (WLAN)	IEEE 802.16 (WWAN)
	ZigBee	6LoWPAN	WirelessHART	ISA100.11a	Bluetooth	WiFi (802.11a,b,g,n,y)	WiMAX (802.16d,16e)
Range	100 m	50 m			100 m	5 km	15 km
Data rate	250-500 Kbps	250 Kbps	250 Kbps	250 Kbps	1 Kbps – 3Kbps	1 Mbps-450 Mbps	75 Mbps
Frequencies (Bandwidth)	2.4 GHz	2.4 GHz	2.4 GHz	2.4 GHz	2.4 GHz	2.4 , 3.7, and 5 GHz	2.3, 2.5, and 3.5 GHz
Network Topology	Star, mesh, cluster tree	Star, mesh, cluster tree IPv6	Star, mesh, cluster tree	Star, mesh, cluster tree	Star	Star, Tree, P2P	Star, Tree, P2P
Applications	Wireless sensors (monitoring and control)	wireless internet, automation and entertainment	Wirelessly process monitoring and control applications	Wireless Systems for Automation	Wireless sensors (monitoring and control)	PC-based Data Acquisition Mobile Internet	Mobile Internet

2.5.1 IEEE 802.15.4 (WSN)

The IEEE 802.15.4 standard defines a low-power/rate wireless personal area network (LR-WPAN). This solution increases network lifetime up to a few years. IEEE 802.15.4 supports a 10-meter communication area with a transfer rate of 250 Kbps at 2.4 GHz. The IEEE 802.15.4 standard defines both the physical (PHY) layer and media access control (MAC) layer specifications, as can be seen in Fig. 12. [22]

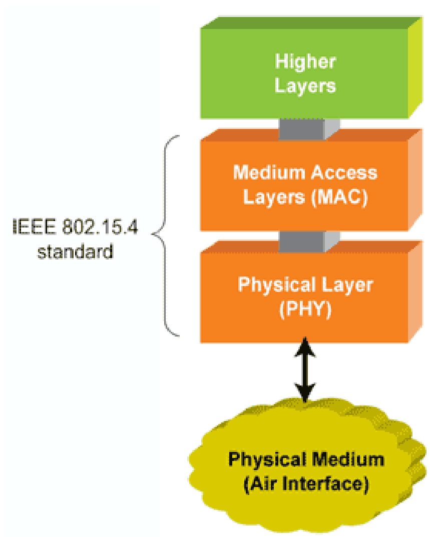

Fig. 12. Structure of IEEE 802.15.4 Protocol Stack

Physical layer

The physical layer is responsible for controlling transmission parameters (see TABLE). The key responsibilities of PHY are:

- activation and deactivation of the radio transceiver
- frequency channel tuning
- carrier sensing
- received signal strength estimation (RSSI & LQI)
- data coding and modulation
- error correction

MAC layer

The key functions of the MAC layer include:

- data framing, and validation of received frames
- device addressing

- channel access management
- device association and disassociation
- sending acknowledgement frames

TABLE 5
Transmission Parameters of IEEE 802.15.4

Frequency MHz	Chip rate	# of channels	Modulation	Data rates (Kbps)	Unlicensed availability
2450	2000	16	O-QPSK	250	Worldwide
915	600	10	BPSK	40, 250	America, Australia
868	300	1	BPSK	20, 100	Europe

At present, the main WSN solutions such as ZigBee, WirelessHART, 6LoWPAN, and ISA-100 are based on this standard, which offers a complete networking solution by developing the remaining upper communication layers.

2.5.1.1 ZigBee

ZigBee is a simple, low cost, low power wireless technology used in LR-PANs embedded applications [23] [24]. ZigBee provides the network layer (NWK) and the framework for the application layer. The MAC sub-layer and lower layers are based on the IEEE 802.15.4 standards (see Fig. 13).

The ZigBee network layer (NWK) supports star, tree, and mesh topologies. It is utilized in three types of devices: ZigBee coordinator, ZigBee routers, and end devices.

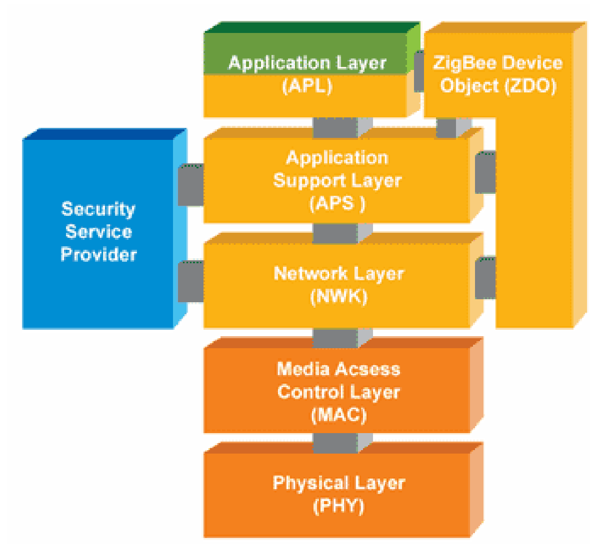

Fig. 13. ZigBee Protocol Stack

2.5.1.2 WirelessHART

Based on IEEE 802.15.4 standard highway addressable remote transducer (HART) foundation, WirelessHART was developed for low-power 2.4 GHz operation. Similar to ZigBee, WirelessHART specifies four principal devices (Fig. 14): network manager, gateways, field devices and handhelds, as well as adapters which allow existing HART field devices to be integrated into the network [13] [21] [25] [26].

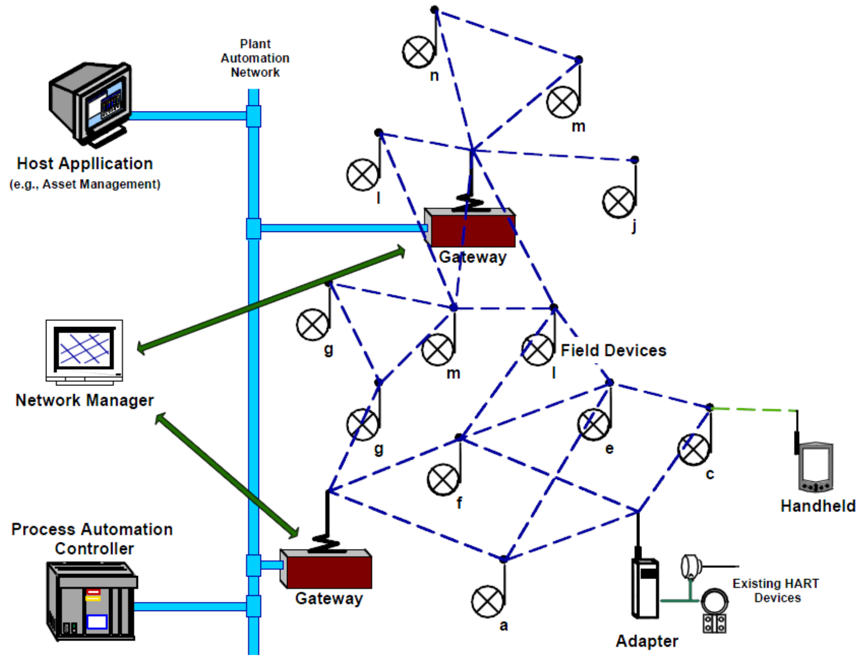

Fig. 14. Elements of a Typical WirelessHART Installation

2.5.1.3 6LoWPAN

6LoWPAN is the abbreviation of IPv6 over Low power Wireless Personal Area Networks, in which standard IPv6 packets are enabled to communicate over an IEEE 802.15.4-based network. Using 6LoWPAN, low power devices have all the benefits of IP communication and management. It is targeted at wireless IP networking applications in home, office and factory environments [27].

2.5.1.4 ISA100.11a

The core wireless technology employed by ISA100.11a is IEEE 802.15.4, which sorts the 2.4 GHz unlicensed band into 16 channels [28].

As shown in Fig. 15, the network layer is based on IETF RFC4944 (6LoWPAN):

- IP connectivity through compressed IPv6 and UDP packets
- addressing scheme:
    - EUI-64 (64 bits)
    - IPv6 (128 bits)
    - Short address (16 bits IEEE 802.15.4)

The entire ISA100.11a stack is constructed employing widely accepted and proven industry standards.

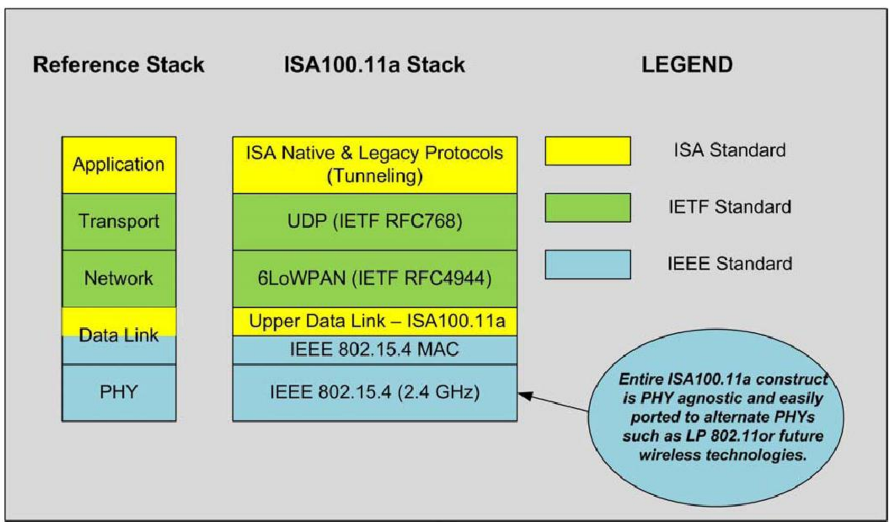

Fig. 15. ISA100.11a Stack

2.6 Operating Systems and Prototypes

The operating system here can represent the low-level functions of sensor nodes by means of a clear interface to the external world, for instance processor management, memory management, device management, scheduling policies or threading.

Many vendors offer a programmable middleware layer (Fig. 16) such as a driver API or standard communication protocol, which is software residing between the operating system and the application.

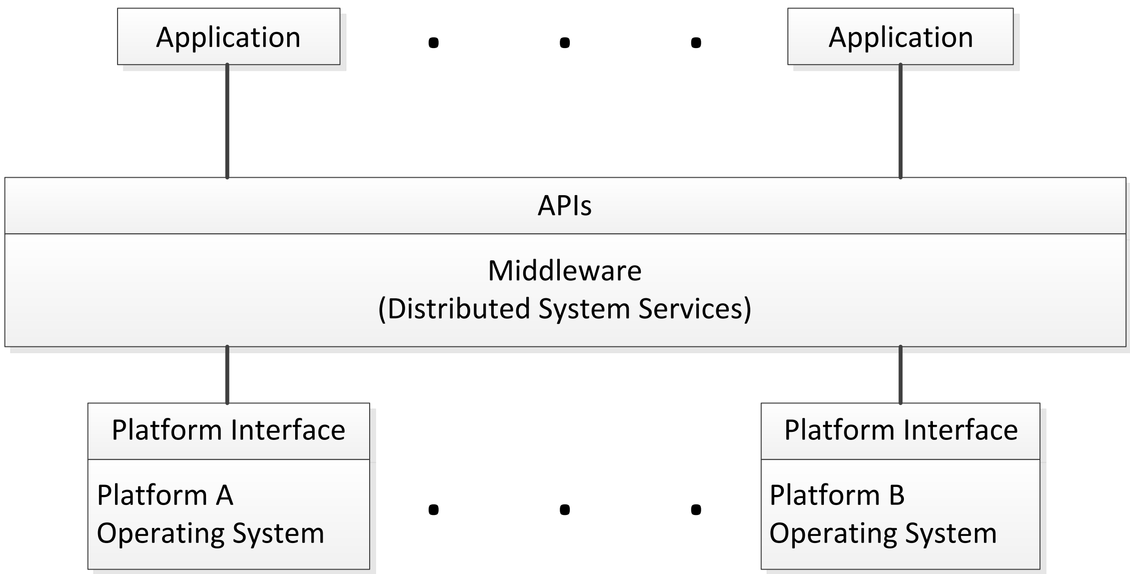

Fig. 16. Middleware Implementation

There are many important features that can be taken into account while designing, comparing and classifying WSN operating systems, which include architecture, execution model, reprogramming, scheduling and power management, as can be seen Fig. 17. WSN classification is based on the above-mentioned features.

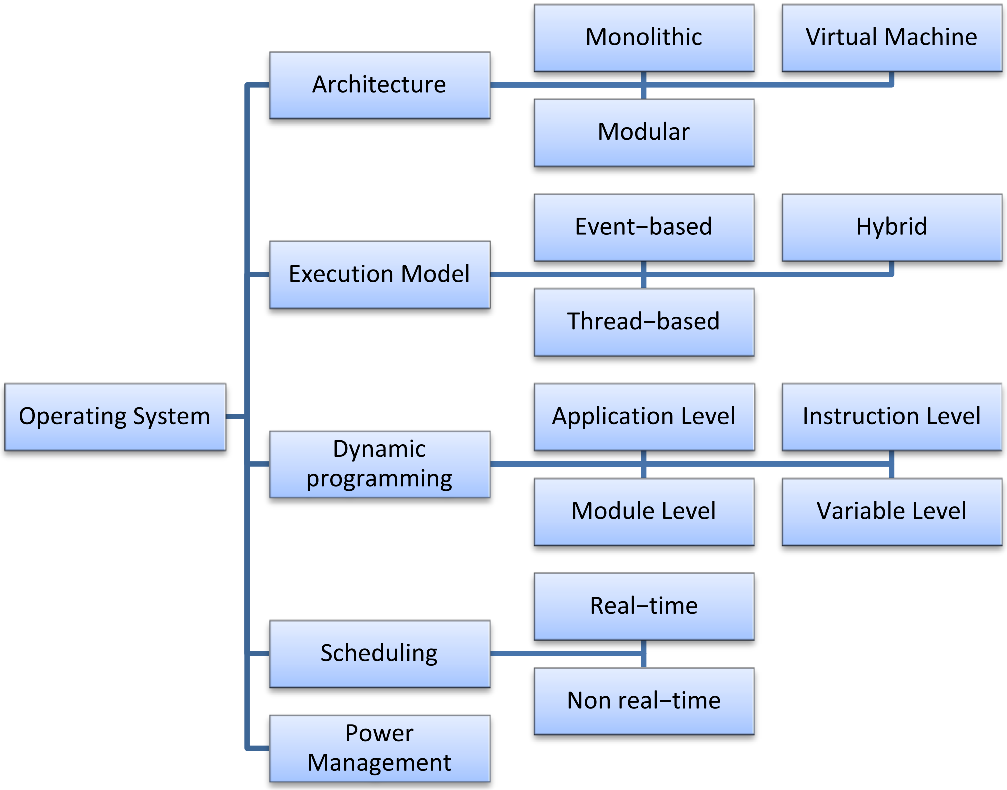

Fig. 17. WSN Operating System Features

- **Architecture** is the organization of the OS and the applications that are executed. It includes:
    - **Monolithic:** OS and the applications are in the same program.
    - **Modular:** OS and application are in separated layers.
    - **Virtual machine-based:** OS includes application components.
- **Execution Model:** The most common execution models are event-based and thread-based. In the past, applications generated events and dealt with them sequentially. There is also a hybrid, with an execution model based on a state machine.

- **Reprogramming** refers to the possibility of modifying the software that is executed in the OS. This modification can be done to the application level, module level, instruction level or variable level.
- **Scheduling:** Here the processor is shared by the different tasks. The most common scheduling approaches in WSNs are periodical (non-real time), non-periodical (real time) and critical/non-critical. For example, sample collecting can be a periodical event, while alarm generation is non-periodical.
- **Power Management:** Operating system can be used to enforce an optimal method of utilizing energy. The components that can be controlled to save power are the processor and the radio.

2.6.1 Available OS

TABLE lists some of the available operating systems [8] [29]. The most well-known OSs are TinyOS and Contiki. Microsoft has also started working in this field using the Microsoft .NET Micro OS. In this survey the available WSN OSs are listed and the most common OSs are explained.

TABLE 6
WSN Available Operating Systems

Operating System	Architecture	Execution Model	Scheduling
TinyOS	Monolithic	Event-based	Non Real-time
Contiki	Modular	Hybrid	Non Real-time
FreeRTOS	Modular	Event-based	Real-time
RETOS	Monolithic	Thread-based	Priority-based
MicroC/OS II	Monolithic	Thread-based	Real-time

VMSTAR	VM		Non Real-time
Nano-Qplus	Modular	Thread-based	Real-time
SOS	Modular	Event-based	Non Real-time
Mate	VM	Event-based	Non Real-time
Magnet	Monolithic		Non Real-time
MANTIS	Modular	Thread-based	Non Real-time
OSPM		Event-based	
kOS	Modular	Hybrid	Non Real-time
EYES OS		Event-based	Non Real-time
SenOS		State-driven	Non Real-time
PicOS	Modular	Event-based	Real-time
Microsoft .NET Micro	VM	Thread-based	Real-time
CORMOS	Modular	Event-based	Real-time

2.6.1.1 TinyOS

TinyOS, which is an open source and monolithic-based OS, is the most widely used operating system in WSN [30]. It began as a partnership between the University of California, Berkeley, Intel Research and Crossbow Technology, and has grown into an international group known as the TinyOS Alliance. Fig. 18 shows simplified TinyOS architecture [31].

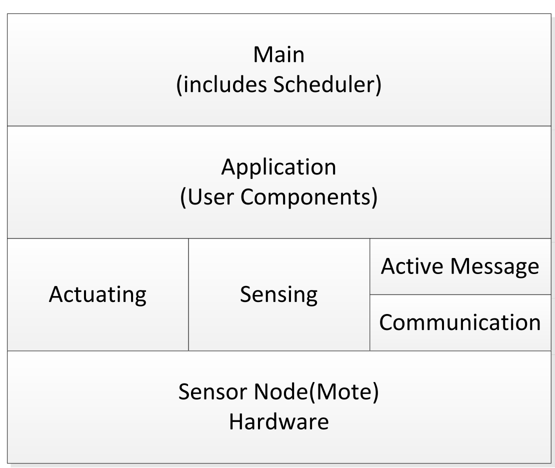

Fig. 18. Simplified TinyOS Architecture

TinyOS features:

- written in “nesC” language, a dialect of ‘C’ language
- no kernel: direct hardware manipulation
- no process management: one process at a time, 255 maximum tasks
- no virtual memory: single linear physical address space
- no user interface
- extremely passive vigilance (power saving)
- supports multi-hop communication
- simulators: TOSSIM and PowerTOSSIM

2.6.1.2 Contiki:

Contiki is an open-source, very portable, multi-tasking operating system for low-power embedded systems. Contiki enables all OS standard functions such as threads (Fig. 20), timers, random number generations, file system and command line shell (Fig. 19).

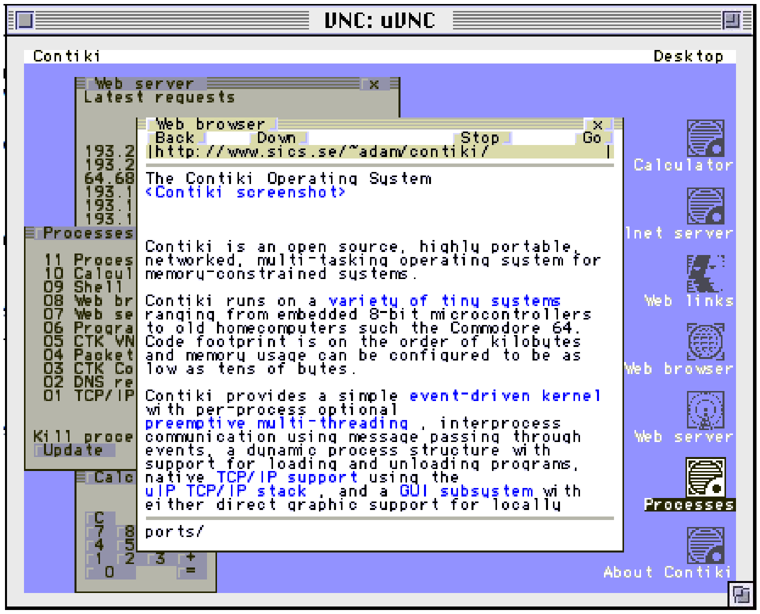

Fig. 19. Screenshot of the VNC Server Running on the Atmel AVR Port of Contiki

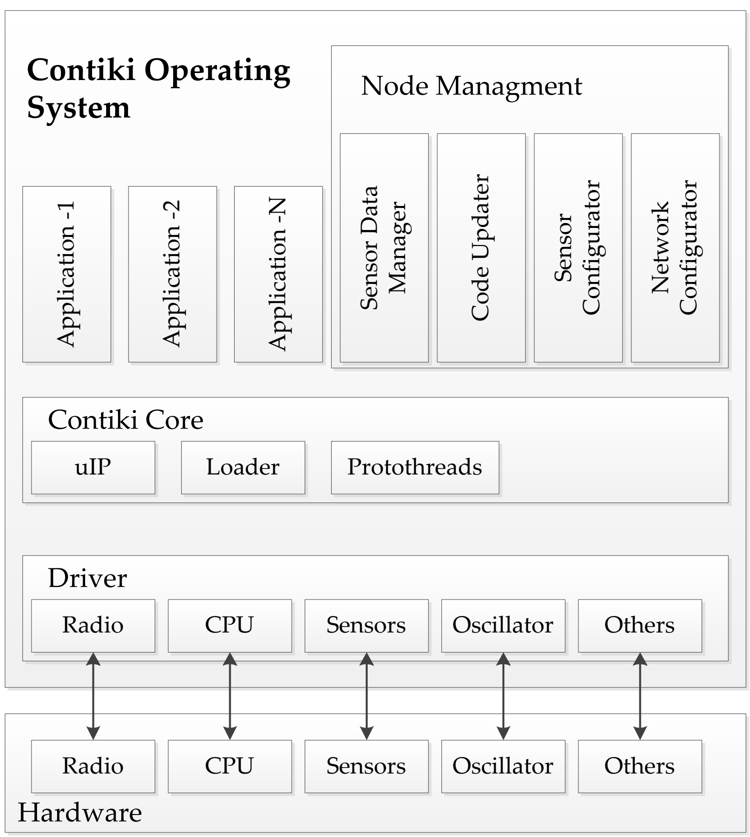

Fig. 20. Simplified Contiki Kernel Architecture

Contiki features:

- Event-driven kernel
- proto-thread support: processes use lightweight proto-threads
- preemptive multi-threading support
- supports a GUI
- many important platforms, such as Tmote Sky and (ESB) Embedded Sensor Board, TelosB and MSP430, use Contiki
- simulator: COOJA
- written in ‘C’ language

2.6.1.3 MANTIS OS

The MANTIS OS is an open-source operating system, with energy-efficient multithreading functionality for multimodal networks of in-situ sensors (MANTIS) nodes. As seen in Fig. 21, user-level threads play a significant role in the MANTIS OS.

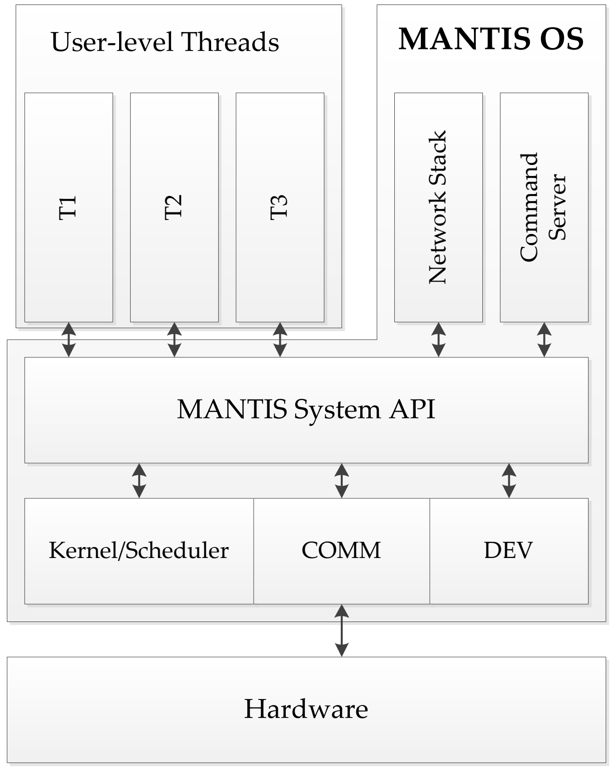

Fig. 21. Simplified MANTIS OS Kernel Architecture

MANTIS Features:

- preemptive multi-threading support
- power-efficient scheduler
- dynamic reprogramming and remote login
- cross-platform support across PC's, PDAs and micro sensor platforms
- written in 'C' language

2.6.1.4 SOS

The main advantage of SOS is the ability to achieve a high reprogramming capability to add, modify and delete software modules at runtime.

SOS Features:

- static kernel interface: to access timers, memory, sensors, and actuators (see Fig. 22)
- priority-based asynchronous messaging communication module
- power management included with special APIs
- built-in simulation framework
- written in ‘C’ language

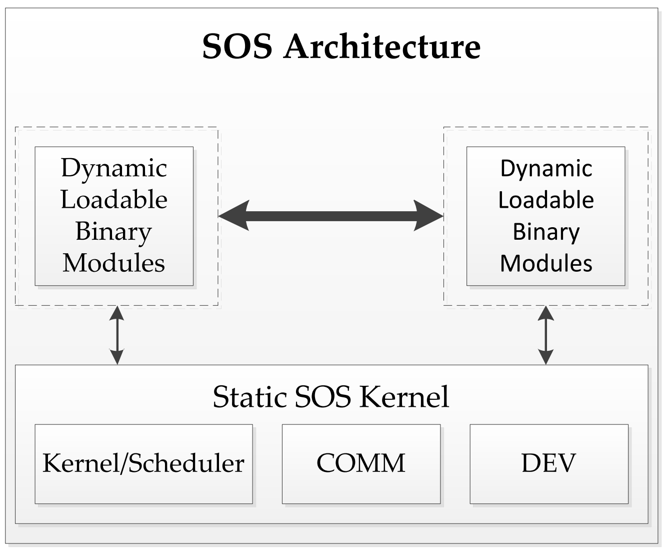

Fig. 22. SOS Architecture

2.6.1.5 Microsoft .NET Micro

Microsoft .NET Micro is a solution from Microsoft for embedded platforms with constrained resources. It offers the advantages of .NET tools for platforms of this type, such as Visual Studio .Net [32].

Fig. 23 shows end-to-end .NET Micro Framework architecture.

Microsoft .NET Micro features:

- the smallest .NET footprint yet developed, able to run from ROM or flash memory
- not real-time, but can be hosted on a real-time OS as an option
- can develop full solutions using .NET, including devices, servers, and the cloud
- Full Visual Studio integration, including live debugging of code running

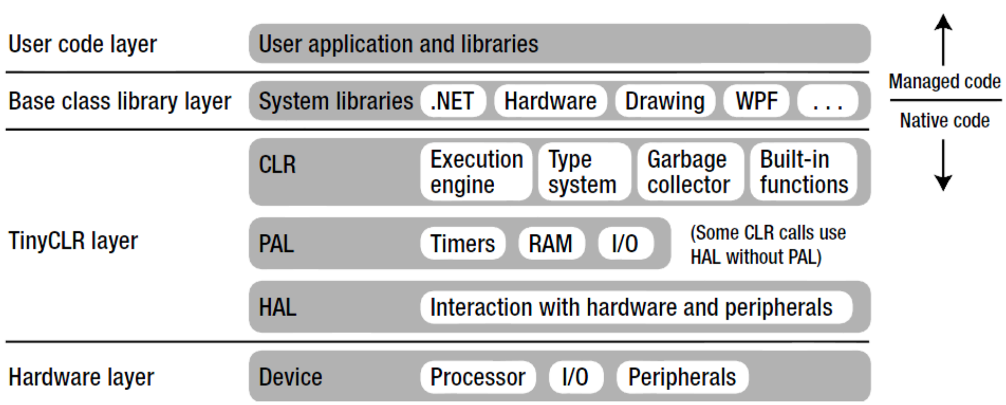

Fig. 23. Layered Architecture of the .NET Micro Framework

The main disadvantage of the .Net Micro frame is that it only supports the IMote 2.0 platform, while (as per TABLE) other open source operating systems can be applied over a wider number of platforms.

TABLE 7
Operating System and Supported Platforms

Operating System	Supported Platform
TinyOS	BTnode, EyesIF X v1, EyesIF X v2, IMote, IMote 1.0, IMote 2.0, Iris, KMote, Mica, Mica2, MicaZ, Rene, SenseNode, TelosB, T-Mote Sky, Shimmer
Contiki	T-Mote Sky, TelosB, avr MCU, MSP430 MCU, x86, 6502
Mantis OS	Mica2, MicaZ, Nymph, TelosB
SOS	XYZ, T-Mote Sky, KMote, Mica2, MicaZ, TelosB, avrora, Protosb, Cricket, Cyclops, emu
Microsoft .NET Micro	IMote 2.0

2.7 WSN Simulators

Using simulators is necessary for any research in the field of wireless communication. In this section I will list the most known simulation environments used to simulate WSN. I will focus on free, open-source, simulation tools, which can be categorized to specific framework simulator and general simulation tools [2] [33] [34] [35].

2.7.1 Specific Framework Simulator

We already mentioned these simulators when we talked about operating system and frameworks. Those specific simulators are designed for specific product or framework. Such as:

- TOSSIM simulator using nesC code on TinyOS/MICA nodes.
- COOJA simulator writing in C language for Contiki OS.
- EmSim on developed using EmStar on microservers Ad-hoc systems.
- ATEMU emulator for AVR processor used in the MICA platform.

2.7.2 General Simulation Tool

A General Simulation is a framework independent simulator that can simulate general WSN scenarios. I will focus here in Omnet++ simulator which is the base of many WSN active simulator packages.

2.7.2.1 NS-2

Network simulator ns-2 is the most widely used WSN simulator. It began as a general network simulator, and mobile ad-hoc wireless networks model was supported later. It is a generic, discrete event, object-oriented (OO) simulator, written in C++, with an OTcl [1] [34] [36].

Extensions for WSN have been continuously created. For example, one extension adds the concept of a phenomenon to sn-2. A phenomenon describes a physical event such as a chemical cloud or moving vehicle that could be monitored by nearby sensor nodes. In the other hand, Mannasim extension adds a sensing model, several application models, LEACH routing protocol, Mica2 PHY model, etc.

2.7.2.2 J-Sim

J-Sim is a component-based simulation developed totally in Java. It provides real-time process-based simulation [37] [38]. The main benefit of J-sim is its considerable list of supported protocols, including a WSN simulation framework.

2.7.2.3 NCTUns2.0

NCTUns2.0 is a discrete event simulator embedded in the kernel of a UNIX machine. The actual network layer packets are tunneled through virtual interfaces that simulate lower layers and physical devices.

2.7.2.4 JiST/SWANS

JiST/SWANS is a discrete event simulator embeds the simulation engine in the Java byte-code. Models are implemented in Java and compiled.

2.7.2.5 GloMoSim

A simulator for wireless networks built with Parsec. Parsec is a simulation language using C language that adds semantics for creating simulation entities and message communication on a variety of parallel architectures.

2.7.3 Omnet++ (commercial version OMNEST)

OMNET++ [33] (Objective Modular Network Testbed in C++) is an object-oriented modular discrete event network simulator, which mainly used for general wireless network simulation.

OMNeT++ consists of modules that communicate with each other using message passing. Complex compound modules can be built up out of simpler modules.

New versions of OMNeT++ framework include a GUI that can be used to edit network topologies either graphically or in source code.

OPNET's has larger protocol model library than OMNeT++, while its closed source code makes development and problem solving harder.

OMNET++ has the big advantage in the model library and the available model aspects. A significant difference between OMNET++ and OPNET is OPNET models always use fixed topology, while OMNET++'s NED and its graphical editor allow customize topology and parameterized topologies.

2.7.3.1 Castalia Simulation Framework

Castalia [39] [40] is a simulator based on the OMNeT++ platform for Wireless Sensor Networks (WSN), Body Area Networks (BAN) and generally networks of low-power embedded devices. Researchers and developers can use Castalia to test their distributed algorithms and/or protocols in realistic wireless channel and radio models.

The main features of Castalia are:

- advanced channel model based on empirically measured data.
- advanced radio model based on real radios.
- extended sensing modeling provisions
  - highly flexible physical process model
  - sensing device noise, bias, and power consumption
- node clock drift, CPU power consumption.
- MAC and routing protocols available.

2.7.3.2 MiXiM Simulation Framework

MiXiM [41] [42] [43] (mixed simulator) is an open-source OMNeT++/OMNEST modeling framework created for mobile and fixed wireless networks (wireless sensor networks, body area networks, ad-hoc networks, vehicular networks, etc.). MiXiM concentrates on accurate radio channel modeling and the physical and medium access layers: it offers detailed models of radio wave propagation, interference estimation, radio transceiver power consumption and wireless MAC protocols like Zigbee.

2.7.3.3 PAWiS Simulation Framework

PAWiS [44] (Power Aware Wireless Sensor Networks) is a discrete event simulator for Wireless Sensor Networks based on the OMNeT++. It provides functionality to simulate the network nodes with their internal structure, and it simulates also the network between the nodes, and the power consumption of every single node.

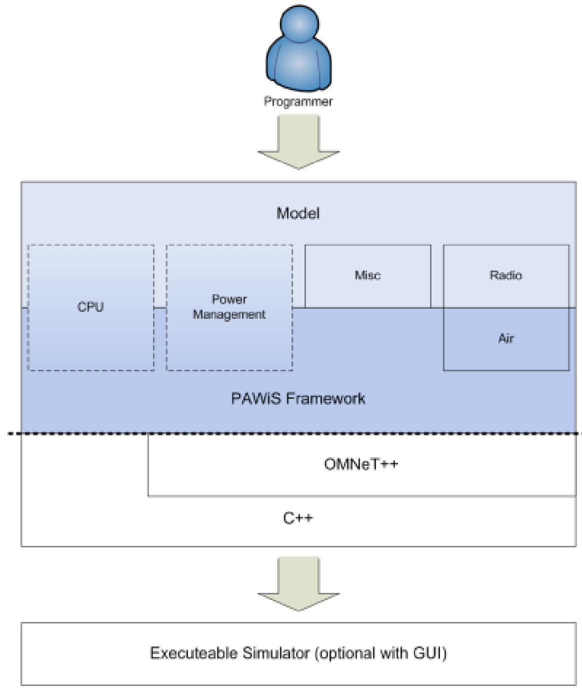

Fig. 24. PAWiS Architecture

A new tool is applied in this simulator to make a data analysis; this tool allows the visualization and analysis of power consumption for each node with its modules and sub-modules as well as event marks (see Fig. 25).

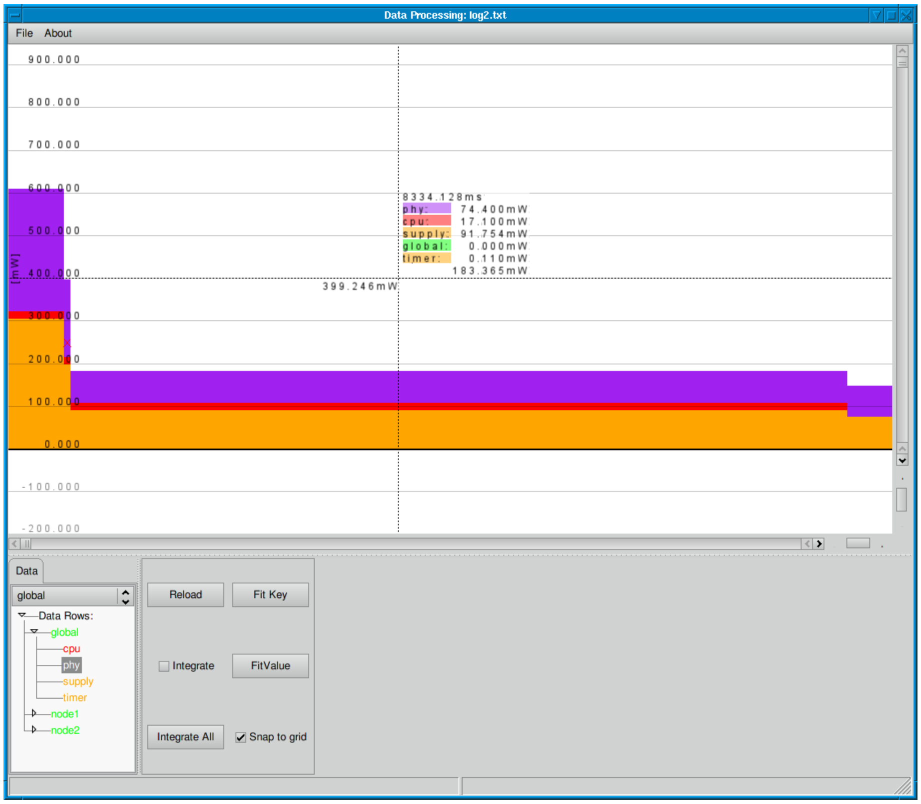

Fig. 25. Data Processing Tool

CHAPTER 3

RELATED WORK

The main goal of WSNs is to detect certain events in the environment. It is important to try to achieve maximum detection accuracy and to minimize false alarms. At the same time, availability of WSN resources and accessibility limitations should be considered. In general, the solution is a trade-off between two factors: accuracy and energy efficiency.

As mentioned earlier, each node has the responsibility of collecting, processing, transmitting and receiving data. The common functions of every detection scheme are that 1) nodes collect observations, and 2) the fusion center (sink) takes the final decision.

There are two traditional detection schemes: the centralized detection scheme and the distributed detection scheme, the methodologies of which will be covered in detail. In the remainder of this section, the hybrid detection scheme, which is built on the two previous schemes, will be described. In any scheme, accuracy of performance and energy efficiency should be defined and compared as metrics of the scheme.

Model Definition: To be able to compare schemes or to propose a scheme, network parameters and assumptions should be defined. It is assumed that all nodes are "independently and identically distributed" (i.i.d.), i.e. the nodes are organized in a tree topology and connected to the FC through a multi-hop route, where nodes also act as hops to receive data from child nodes and forward it to FC without any processing, encryption or encoding. The focus here is only on accuracy and energy consumption, and it is assumed that lower layers are working perfectly and that there is no efficiency problems caused by RF or by packet collisions, i.e. there is no data retransmission.

3.1 Bayesian Decision Theory

3.1.1 Binomial Distribution

Binomial distribution is the discrete probability distribution of the number of successes in a sequence of n independent true/false trials.

The whole wireless sensor network detection process can be regarded as a Bernoulli process, since it shares the same properties:

1. The experiment consists of repeated trials.
2. Trial results are classified as a success or a failure.
3. The probability of success, denoted by p, remains constant from trial to trial.
4. Repeated trials are independent.

A Bernoulli trial can result in a success with probability p and a failure with probability $q=1-p$. Equation (1) gives the probability distribution of the binomial random variable x. The number of successes in n independent trials is:

$$f(x; n, p) = \Pr(x|p) = \binom{n}{x} p^x (1-p)^{n-x} \quad (1)$$

3.1.2 Bayes' Rule

Bayes' Rule is one of the most important rules in probability theory. It is the foundation of Bayesian inference, which is at the heart of calculating the decision accuracy of any scheme. [45]

The probability that two events, A and B, will both occur will be $P(A \cap B) = P(B \cap A) = P(B)P(A|B) = P(A)P(B|A)$. From this formula the main Equations (2) and (3) of Bayes' Rule can be derived:

$$P(A|B) = \frac{P(A)P(B|A)}{P(B)} \quad (2)$$

$$P(A) = \sum_{i=1}^{k} P(B_i \cap A) = \sum_{i=1}^{k} P(B_i)P(A|B_i) \quad (3)$$

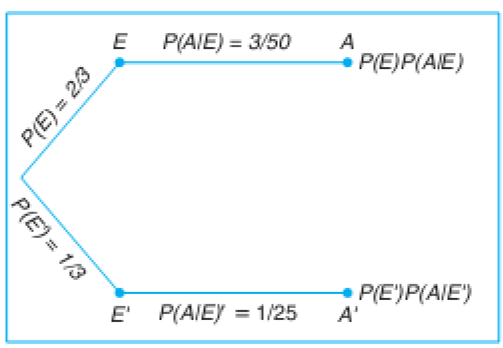

Fig. 26. Conditional Probability Example

From Fig. 26 we find that $P(E \cap A) = P(E)P(A|E) = \left(\frac{2}{3}\right)\left(\frac{3}{50}\right) = 0.04$.

The distribution of *H*, given *D*, which is called the posterior distribution, where *P(D)* is the marginal distribution of *D*, is given by Equation (4),

$$P(H|D) = \frac{P(D|H)\,P(H)}{P(D)} \tag{4}$$

Binary assumption is defined by *H*, which represents event occurrence: $(H = H_1)$ if an event happens and $(H = H_0)$ if not; $\widehat{H}$ is the actual event status: H' is the final decision; *K* is node counts; *D* is the collected samples, *T* is sample count at each node and *L* is the node path length. For each sample S_{ij} where *i* is varied from 1 to *K* and *J* from 1 to T collected in a node *i*, $P\left(S_{ij} = 1\middle|\widehat{H} = H_1\right) = p_1,\ P\left(S_{ij} = 1\middle|\widehat{H} = H_0\right) = p_0$

Bayes Decision: Choose event happened if $P(H_1|D) \geq P(H_0|D)$ otherwise choose not happened.

3.2 Centralized Detection Schemes

In centralized detection schemes a network will have *K* number of nodes. These nodes will collect *T* samples of observations from the environment every specific period, and they will send *T* samples together at the end of the period. At the fusion center $D = [T * K]$ samples will be received.

According to Bayes, a final decision can be calculated as shown in Equation (5):

$$\widehat{H} = \begin{cases} 1, & P(H_1|D) \geq P(H_0|D) \\ 0, & otherwise \end{cases} \tag{5}$$

A decision threshold γ_{FC} can be calculated from the probability mass function of binomial distribution in Equation (1) and Equation (5):

$$P(H_1|D) \geq P(H_0|D) \Rightarrow \frac{P(D|H_1)\,P(H_1)}{P(D)} \geq \frac{P(D|H_0)\,P(H_0)}{P(D)} \Rightarrow \frac{P(D|H_1)}{P(D|H_0)} \geq \frac{P(H_0)}{P(H_1)}$$

$$f(k;m,p) = Pr(k|p) = \binom{n}{k} p^k (1-p)^{n-k} \Rightarrow P(D|H_\theta) = \binom{KT}{D} {p_\theta}^D (1-p_\theta)^{KT-D}$$

$$\frac{\binom{KT}{D} {p_1}^D (1-p_1)^{KT-D}}{\binom{KT}{D} {p_0}^D (1-p_0)^{KT-D}} \geq \frac{(1-p)}{p} \Rightarrow D \geq \frac{ln\frac{1-p}{p} + KT\, ln\frac{1-p_0}{1-p_1}}{ln\frac{p_1(1-p_0)}{p_0(1-p_1)}} = \gamma_{FC} \Rightarrow \widehat{H} = \begin{cases} 1, & D \geq \gamma_{FC} \\ 0, & otherwise \end{cases}$$

$$\frac{ln\frac{1-p}{p} + KT\, ln\frac{1-p_0}{1-p_1}}{ln\frac{p_1(1-p_0)}{p_0(1-p_1)}} = \gamma_{FC} \tag{6}$$

The probability of error can be calculated using the following equation:

$$P_e = \left(p * P_{false\ positive}\right) + ((1-p) * P_{false\ negative}) \Rightarrow$$

$$P_e = p * P[H' = H_1|H_0] + (1-p) * P[H' = H_0|H_1] \Rightarrow$$

$$P_e = p * [1 - P(n \geq \gamma_{FC}|H_1)] + (1-p) * P(n \geq \gamma_{FC}|H_0) \tag{7}$$

To calculate power consumption for the whole network in Equation (8), the following equation can be used, where E = total energy, E_T = transmission energy, E_R = receiving energy and E_P = processing energy:

$$E = E_T + E_R + E_P \Rightarrow E = \sum_{i=1}^{K}(L_i * T * e_t) + \sum_{i=1}^{K}(L_i * T * e_r) \Rightarrow$$

$$E = \sum_{i=1}^{K} (L_i * T * (e_t + e_r)) \tag{8}$$

3.3 Distributed Detection Scheme

In this scheme nodes collect data and make local decisions according to these observations and conclude the event appearance as a 1-bit result. This result is sent to the fusion center to make a final decision according to the collected 1-bit results from all nodes. In this scheme data accuracy between nodes and the fusion center has been lost.

The scheme propose K number of nodes; these nodes will collect T samples of environmental observations with [n_i = number of 1's for node i] and will send a 1-bit local decision every specific period.

At the fusion center, D = [1 * K] samples will be received.

According to Bayes, a local decision can be calculated as shown in Equation (9):

$$\widehat{H_i} = \begin{cases} 1, & P(H_1|n_i) \geq P(H_0|n_i) \\ 0, & otherwise \end{cases} \tag{9}$$

From equation (1) and (5), we can calculate local decision as shown in equation (10):

$$P(H_1|n_i) \geq P(H_0|n_i) \Rightarrow \frac{\mathrm{P}(n_i|\mathrm{H}_1)\,\mathrm{P}(\mathrm{H}_1)}{\mathrm{P}(n_i)} \geq \frac{\mathrm{P}(n_i|\mathrm{H}_0)\,\mathrm{P}(\mathrm{H}_0)}{\mathrm{P}(n_i)} \Rightarrow \frac{\mathrm{P}(n_i|\mathrm{H}_1)}{\mathrm{P}(n_i|\mathrm{H}_0)} \geq \frac{\mathrm{P}(\mathrm{H}_0)}{\mathrm{P}(\mathrm{H}_1)}$$

$$\frac{\binom{T}{n_i} {p_1}^T (1-p_1)^{T-n_i}}{\binom{T}{n_i} {p_0}^T (1-p_0)^{T-n_i}} \geq \frac{(1-\mathrm{p})}{\mathrm{p}} \Rightarrow n_i \geq \frac{\ln\frac{1-p}{p} + T\ln\frac{1-p_0}{1-p_1}}{\ln\frac{p_1(1-p_0)}{p_0(1-p_1)}} = \gamma_{local} \Rightarrow$$

$$\widehat{H} = \begin{cases} 1, & n_i \geq \gamma_{local} \\ 0, & otherwise \end{cases} \quad (10)$$

For final decision we collect b = total 1's if local decision $\widehat{H} = \begin{cases} 1, & P(H_1|b) \geq P(H_0|b) \\ 0, & otherwise \end{cases}$

From equation (1) and (5), we can calculate final decision as shown in equation (11):

$$, P(H_1|b) \geq P(H_0|b) \Rightarrow \frac{\mathrm{P(b|H_1)\ P(H_1)}}{\mathrm{P(b)}} \geq \frac{\mathrm{P(b|H_0)\ P(H_0)}}{\mathrm{P(b)}} \Rightarrow \frac{\mathrm{P(b|H_1)}}{\mathrm{P(b|H_0)}} \geq \frac{\mathrm{P(H_0)}}{\mathrm{P(H_1)}}$$

$$, P_D = \sum_{i=\gamma_{local}}^{T} \binom{T}{i} {p_1}^T (1 - p_1)^{T-i} \ \& \ P_F = \sum_{i=\gamma_{local}}^{T} \binom{T}{i} {p_0}^T (1 - p_0)^{T-i}$$

$$, \frac{\binom{K}{b} {p_\mathrm{D}}^K (1-p_\mathrm{D})^{K-b}}{\binom{K}{b} {p_\mathrm{F}}^K (1-p_\mathrm{F})^{K-b}} \geq \frac{(1-\mathrm{p})}{\mathrm{p}} \Rightarrow b \geq \frac{\ln\frac{1-p}{p} + K \ln\frac{1-p_F}{1-p_D}}{\ln\frac{p_D(1-p_F)}{p_F(1-p_D)}} = \gamma_{FC} \Rightarrow$$

$$\widehat{H} = \begin{cases} 1, & b \geq \gamma_{FC} \\ 0, & otherwise \end{cases} \quad (11)$$

The probability of error can be calculated using Equation (12) below:

$$P_e = \left(p * P_{false\ positive}\right) + ((1 - p) * P_{false\ negative}) \Rightarrow$$

$$P_e = p * [1 - P(b \geq \gamma_{FC}|H_1)] + (1 - p) * P(b \geq \gamma_{FC}|H_0) \quad (12)$$

To calculate power consumption for the whole network Equation (13) can be used, where E = total energy, E_T = transmission energy, E_R = receiving energy and E_P = processing energy:

$$E = E_T + E_R + E_P \Rightarrow E = \sum_{i=1}^{K} (L_i * 1 * e_t) + \sum_{i=1}^{K} (L_i * 1 * e_r) + \sum_{i=1}^{K} \left(T * e_p\right) \Rightarrow$$

$$E = \sum_{i=1}^{K} (L_i * (e_t + e_r)) + \sum_{i=1}^{K} (T * e_p) \qquad (13)$$

3.4 Hybrid Detection Scheme

Neither the centralized nor the distributed detection scheme is flexible enough for designers to choose between detection accuracy and energy consumption. Because of that, Lige Yu et al [46]proposed a hybrid scheme that balances detection accuracy and total energy consumption. According to a defined level of accuracy, the nodes will vary between sending all collected data and sending a 1-bit result. Thus, such schemes attempt to balance accuracy and energy consumption

In this scheme, assume there are K number of nodes. These nodes will collect T samples of environmental observations with [n_i = number of 1's for node i]. There will be upper and lower bounds N_0 and N_1, where $0 \leq N_0 < N_1 \leq T$. The node result will be 0 if the number of 1's are less than or equal to N_0. In other words, if the number of 0's collected is greater than or equal to T-N_0, 1 will be sent if the number of 1's are greater than or equal to N_1. Otherwise all the collected data will be sent, as shown in Equation (14):

$$Result_i = \begin{cases} 1, & n_i \geq N_1 \\ n_i, & N_0 < n_i < N_1 \\ 0, & n_i \leq N_0 \end{cases} \qquad (14)$$

From Equation (14), assume that out of K sensor nodes, t nodes send 1's, s nodes send 0's and $K-s-t$ nodes send all their observations. The total data sent, Ω, will be:

$$\Omega = \{1, \ldots, 1; n_1, \ldots, n_{K-s-t}; 0, \ldots, 0; \}$$

From Equation (14) we can derive the following probability:

$P[b=0|H_\theta] = \sum_{i=0}^{N_0} \binom{T}{i} p_\theta{}^T (1-p_\theta)^{T-i}$ & $P[b=1|H_\theta] = \sum_{i=N_1}^{T} \binom{T}{i} p_\theta{}^T (1-p_\theta)^{T-i}$

$$PD = \binom{K}{s}\binom{K-s}{t}(P[b=0|H_1])^s (P[b=1|H_1])^t (P[n_i=1|H_1])^{k-s-t} \quad (15)$$

$$PF = \binom{K}{s}\binom{K-s}{t}(P[b=0|H_0])^s (P[b=1|H_0])^t (P[n_i=1|H_0])^{k-s-t} \quad (16)$$

Following from Equations (15) and (16), the final probability of error can be determined using Equation (17):

$$P_e = p * [1-PD] + (1-p) \times PD \quad (17)$$

To calculate power consumption for the whole network Equation (13) can be used, where E = total energy, E_T = transmission energy, E_R = receiving energy and E_P = processing energy:

$E = E_T + E_R + E_P \Rightarrow E = [\sum_{i=1}^{s+t}(L_i * 1 * e_t) + \sum_{i=1}^{K-s-t}(L_i * T * e_t)] + [\sum_{i=1}^{s+t}(L_i * 1 * e_r) + \sum_{i=1}^{K-s-t}(L_i * T * e_r)] + \left[\sum_{i=1}^{K}\left(T * e_p\right)\right] \Rightarrow$

$$E = \sum_{i=1}^{s+t}(L_i * (e_t + e_r)) + \sum_{i=1}^{K-s-t}(L_i * T * (e_t + e_r)) + \sum_{i=1}^{K}\left(T * e_p\right) \quad (18)$$

CHAPTER 4

RESEARCH METHODOLOGY

4.1 Problem Statement

Monitoring environmental changes and detecting specified events are the main functions of wireless sensor networks. WSNs achieve this by using centralized fusion centers (sinks) and many distributed sensors. These sensors continuously sense changes and send the observations to a centralized unit. The centralized unit decides if an event is initiated or not.

Since power resources for sensors is limited in WSNs, it is important to reduce nodes' power consumption while maintaining acceptable detection accuracy according to application requirements.

Processing, transmitting, sensing consumption or all of these should be reduced in order to reduce power consumption.

Substantial research has been done on the reduction of power consumption. One scheme suggests collecting observations, processing them locally in the distributed node and sending optimized data instead of sending all the observations. This means that the data sent from nodes – and forwarded as well – will be reduced. On the other hand, there will be a significant loss of data and accuracy.

In this thesis, I will attempt to enhance both processing and transmission functionality. I will try to define a scheme that balances reduced transmission action with detection accuracy. The balance will be achieved by utilizing both distributed and centralized processing.

4.2 System Model (TELOS)

In order to evaluate and develop our scheme, the behavior of WSN sensors should be understood, and a power consumption model should be defined. [47]

For this purpose the typical operation conditions of TelosB (Fig. 27) have been selected as a basis for our power model. [48] [49]

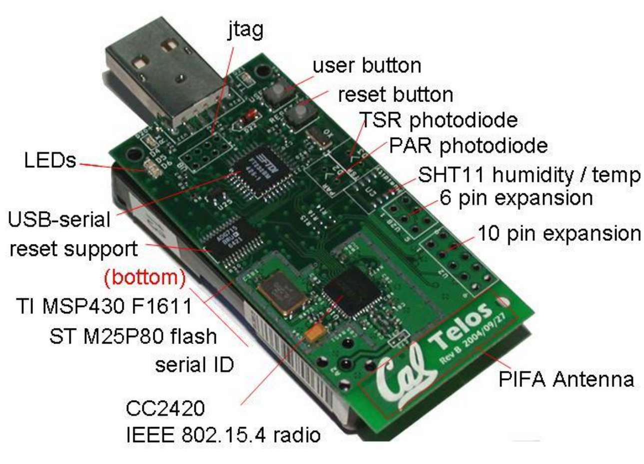

Fig. 27. TelosB by the University of California

I use the datasheet of Telos working at 250kbps and 2.4GHz (Ultra low power IEEE 802.15.4 compliant wireless sensor module), which is one of the most widely known WSN nodes, and I use the power details from TABLE :

TABLE 8
Telos Typical Operation Conditions [50]

	MIN	NOM	MAX	UNIT
Supply voltage	2.1		3.6	V
Supply voltage during flash memory programming	2.7		3.6	V
Operating free air temperature	-40		85	°C
Current Consumption: MCU on, Radio RX		21.8	23	mA
Current Consumption: MCU on, Radio TX		19.5	21	mA
Current Consumption: MCU on, Radio off		1800	2400	µA
Current Consumption: MCU idle, Radio off		54.5	1200	µA
Current Consumption: MCU standby		5.1	21.0	µA

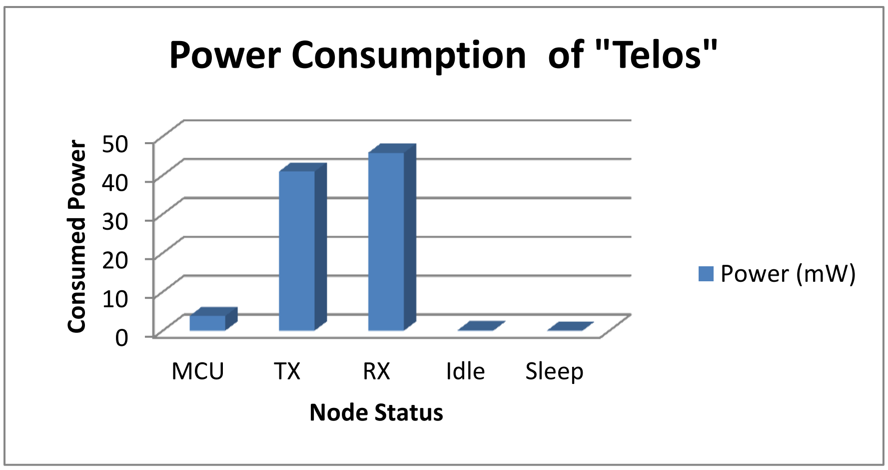

Fig. 28. Power Consumption of "Telos"

From the power consumption model in Fig. 28, it can be concluded that transmission is responsible for a large amount of node power consumption.

So far only the behavior of a single node has been discussed. Now the power consumption of all the nodes in the network should be addressed.

The formula for calculating the overall power consumed while sending a single bit from source (sensor node) to destination (fusion center) through N hop nodes equals the energy required to send and receive this bit between all of the N hops, where E = total energy, E_T = sending energy and E_R = receiving energy:

Energy consumed per node: $E = E_T + E_R$ ➔ $E = (N * e_t) + ([N-1] * e_r)$

So, for the special case where a node is directly connected to the FC, energy used = $e_t + e_r$,

For instance, if hops count = 10, energy used will = 10 $e_t + 9\ e_r$.

Thus, the length of a route affects the overall power consumed in the network, and the network life time as well.

4.3 Power Calculation:

The TelosB model will be used to calculate power consumption. The same calculation methodology will be employed in our simulation [4].

- MCU current consumption: 2.4 mA.
- Rx current consumption: 23 mA.
- Tx current consumption: 21 mA.

4.3.1 Processing Energy:

The MSP430 [51] is running at a clock rate of 8MHz, Instruction Cycle Time = 200 ns, so every cycle we can finish $\frac{Cycle\ time}{instruction\ time} = \frac{8*10^{-6}}{200*10^{-9}} = 40\ \text{instruciton/Cycle}$

Let's assume that we need 2,000 instructions for the measurement, for data processing and for preparing a packet for transmission over the network. This would result in $\frac{2000}{40*8*10^{6}} = 6.25 * 10^{-6}\ \text{s}$, the amount of energy consumed per sample at a current of 2.4mA would be:

$$E_P = 2.4 * 6.25 * 10^{-6} = 15 * 10^{-6}\ \text{mA/s} \qquad (19)$$

4.3.2 Transmission Energy:

Using CC2420, sending data takes place at a speed of 250 kbps. If one node collects from a local sensor or from all the children 1 bit of data to be forwarded to the parent node, power consumption can be calculated as in Equations (19) and (20):

$$\frac{1}{250 * 10^{3}} = 4 * 10^{-6}\ \text{s}$$

$$E_R = 23 * 4 * 10^{-6} = 92 * 10^{-6}\ \text{mA/bit} \qquad (20)$$

$$E_T = 21 * 4 * 10^{-6} = 84 * 10^{-6}\ \text{mA/bit} \qquad (21)$$

4.4 Simulation Network Design

In order to derive more realistic data from our simulation model, we will apply TelosB properties in our network, with maximum coverage = 100 m. A total of *N* sensors are randomly deployed in the region of interest (ROI) which is a square area of a^2 I have selected *a* to be equal to 300 m; hence the ROI = 0.09 km2. The locations of sensors are unknown before deployment time. However, it is known that all sensors are i.i.d., and every sensor locations *(x, y)* will follow a uniform distribution in the ROI.

Fig. 29 shows an example of random sensor deployment for WSNs.

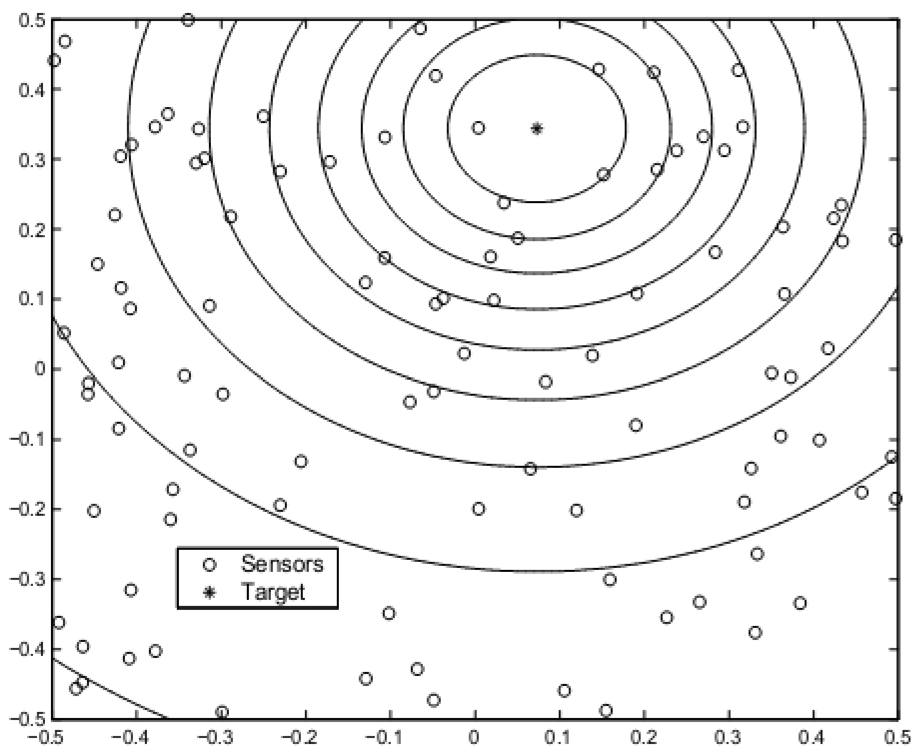

Fig. 29. An Example of Sensor Deployment [52]

CHAPTER 5

ENHANCED HYBRID DETECTION SCHEME

The aim of this thesis is to present an enhanced hybrid detection scheme.

Since Lige Yu et al in the hybrid scheme adjusts the behavior of the network to vary between centralized and distributed schemes, the author establishes the N_0 and N_1 parameters to define that behavior, and Y is the number of observations equal 1.

Lige Yu et al defines all nodes as having the same behavior with the same probability, regardless of node position and event source. Since the nodes have different route lengths, there will be differences in their power consumption. A hybrid schemes is not efficient in this respect.

In this thesis we are proposing to enhance the hybrid scheme by dynamically choosing the N_0, N_1 parameter instead of its being static.

In the hybrid scheme if Y is between N_0 and N_1 the node will act as centralized; otherwise it will act as distributed, as shown in Fig. 30. For the special case N_1-$N_0 \leq 1$ the node will always act as distributed, and the node will act more centralized if N_1-N_0 becomes larger (until $N_0=0$ and $N_1 = MAX$).

Fig. 30. Node Behaviour Depends on N0 and N1

However, N_0 and N_1 can be made dynamic, with a preference for a distributed orientation for nodes with longer route paths; for the remaining nodes it can remain more centralized, as in the original hybrid scheme, according to the requirements of the application.

For every sensor S is varied from 1 to K we will assign specific $\{N_{0i}, N_{1i}\}$. These sensors will be classified according to the route path.

$$S_i = \begin{cases} 1, & n_i \geq N_{1i} \\ n_i, & N_{0i} < n_i < N_{1i} \\ 0, & n_i \leq N_{0i} \end{cases} \qquad (22)$$

Since $0 \leq N_0 < N_1 \leq T$, we will have a finite number of combinations for N_0 and N_1. These combinations - or pair of them - should be mapped to all sensors, depending on sensor path weight.

The above mapping is a normal n-to-one mapping problem, which in our case can be solved experimentally by testing it in different deployed wireless sensor applications.

5.1 Network Deployment

Similar to most wireless sensor networks, the FC is deployed in an accessible location, while the sensor nodes are deployed randomly in the targeted area. This can be considered as a mesh network (Fig. 31). After deploying the nodes, discovering the network is the first action to be taken. The FC will broadcast discovery packets to all nodes in the coverage. These nodes will then broadcast to further nodes until all the nodes are covered and routing paths; next hub and node configuration are defined.

To simplify the problem our nodes will discover paths to the FC based on SPF (shortest path first) and select subsequent hubs to forward received packets to. This information is crucial.

Since the nodes have already collected the broadcast packets, which includes source, destination and path, the nodes will be able to select their configuration locally based on the path length.

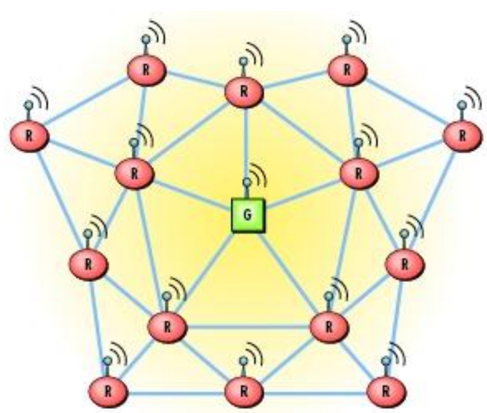

Fig. 31. Sample Sensor Mesh Network

5.2 Nodes Configuration

For each node there are fixed configurations such as Tx/Rx power, frequencies and calculation process. Some parameters need to be configured by the manufacturer or the user in order to optimize network efficiency. On the other hand, some parameters may be configured by the network itself.

In our scheme, every node will have certain parameters that are defined during the manufacturing phase or by the developer during network deployment, i.e. sampling rate *(T)*, CP_{min} and CP_{max}.

CP is the probability that the system will work as centralized, which can be calculated from N_0 and N_1. In our scheme each node will select N_0 and N_1 depending on its path length. N_0 and N_1 should generate *CP* where $CP_{Min} \leq CP_{N0,N1} \leq CP_{Max}$..

From those inputs {*T*, *p0*, *p1*, *max(path_length)*, CP_{min}, CP_{max}} each node will be able to calculate the N_0 and N_1 that satisfy application requirements.

From *T* we can find (N_0,N_1) combinations = 2^T, which is our ROT. Every node should be able to map the proper (N_0,N_1) from both (CP_{min}, CP_{max}) and *max(path_length)*, as can be seen in Fig. 32.

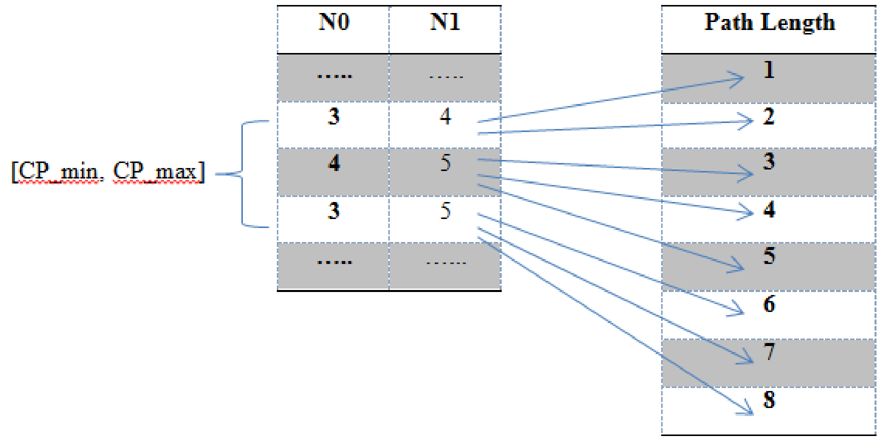

Fig. 32. Mapping (N_0,N_1) Combinations to Available Path Length

We can map these ranges using simple 1-to-many mapping technique, where we index the initial range and assign the remaining ones equally to the next range, as applied in Algorithm 1.

Algorithm 1: Local N_0 & N_1 Selection

Define $\{CP_{min}, CP_{max}\}$

for each $\{N_0,N_1\}$

$\quad$ $CP_i = binCDF(N_{1i}-1,T,P[S=1]) - binCDF(N_{0i},T,P[S=1]);$

$\quad$ **if** $CP_{min} < CP_i < CP_{max}$

$\quad\quad$ add CP_i to CP;

$\quad$ **end if**

end for

$seed = \lceil count(Path)/count(CP) \rceil$

for i = 1 ➔ count(*Path*)

$\quad$ $cp\_seed = min(ceil(i/seed),size(CP,1))$

$\quad$ $CP_i = CP(cp\_seed);$

end for

5.3 N_0, N_1 Selection

To be able to select N_0 and N_1 at each node the problem is divided into simpler problems, as follows:

1. Define all combination of N_0 and N_1 and calculate the equivalent *CP*.
2. Define valid N_0 and N_1 based on CP_{min} and CP_{max}.
3. Define Max path length, and map valid (N_0, N_1) to every path zone.

5.3.1 All N_0, N_1 combinations:

A selection of N_0 and N_1 will define when the node will act as more centralized or more distributed. For T collected samples we will have $0 \leq N_0 < N_1 \leq T$. We can have *C* different combinations for those N_0 and N_1, where $C = T!/((T-2)!\ 2!)$.

For each combination we can calculate its CP_{CENT}, and we will have the *C* list of *CP*, which should be sorted according to its values.

CP_{CENT} can be calculated as in Equation (18):

$$CP_{CENT} = binocdf(N_1 - 1) - binocdf(N_0) \qquad (23)$$

where *cdf* is the cumulative distribution function which can be calculated as in Equation (19).

$$y = F(x \mid n,p) = \sum_{i=0}^{x} \binom{n}{i} p^{i} (1-p)^{(n-i)} I_{(0,1,\ldots,n)}(i). \quad (24)$$

For example, for $T=5$, $P[S=1]=0.45$, we generate all possible combinations of (N_0,N_1) as shown in TABLE :

TABLE 9
CP Values for all (N_0,N_1) Combinations

N_0	N_1	CP
0	5	0.93
0	4	0.82
1	5	0.73
1	4	0.61
0	3	0.54
2	5	0.39
1	3	0.34
2	4	0.28
0	2	0.21
3	5	0.11
0	1	0.00
1	2	0.00
2	3	0.00
3	4	0.00
4	5	0.00

5.3.2 Valid N_0 and N_1 combinations:

For example, we can select initial $CP_{min}=0.50$, $CP_{max}=0.90$.

First of all we find the valid (N_0,N_1) combinations. The selection of $CP_{min} = 0.50$, $CP_{max} = 0.90$ gives us 4 valid combinations (TABLE):

TABLE 10

(N_0,N_1) Valid Combination List

CP	N_0	N_1
0.82	0	4
0.73	1	5
0.61	1	4
0.54	0	3

5.3.3 Map (N_0, N_1) combinations to available zones:

In this enhanced scheme all nodes are classified according to path length to *Z* zones. We have here a random WSN where maximum path = 6, which means 6 different zones. Fig. 33 shows an example of WSN nodes classified into zones:

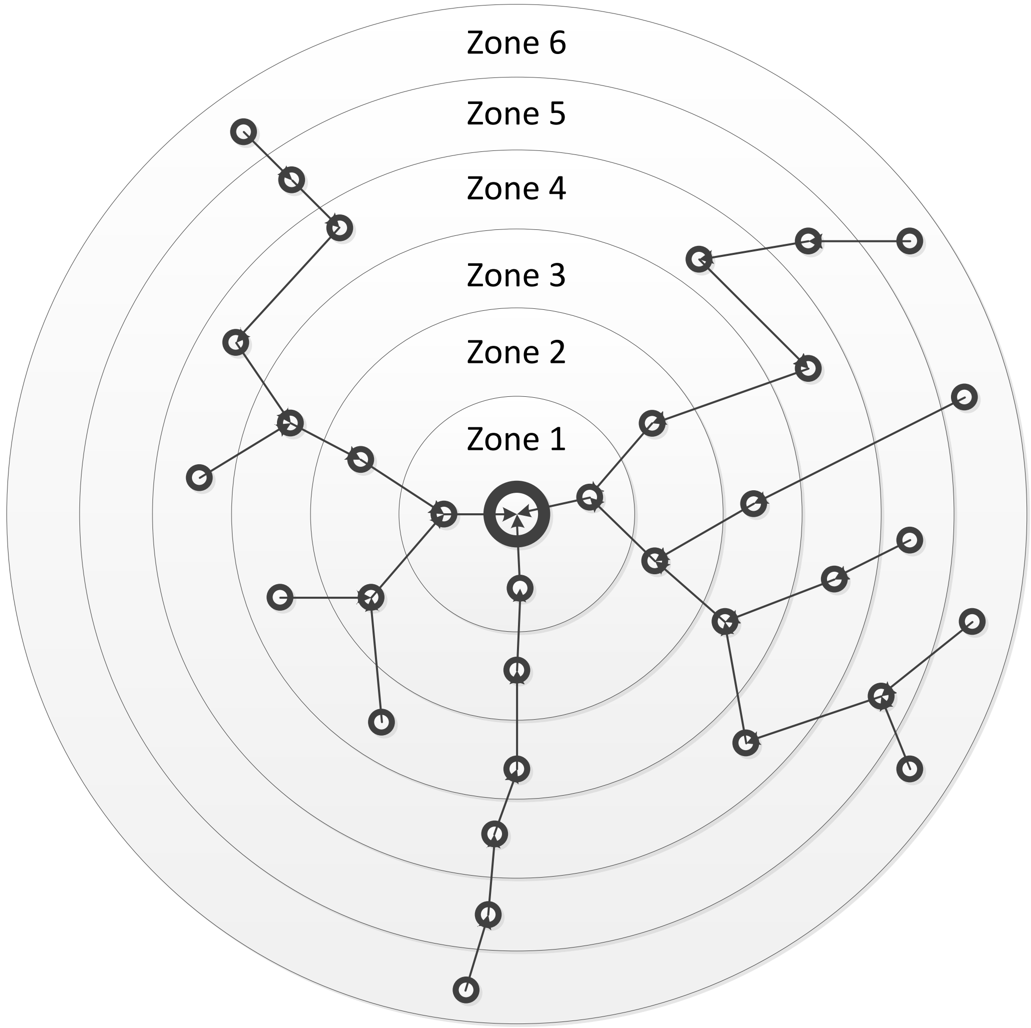

Fig. 33. Network Zones Classification

For our network we will define Minimum CP_{min} and Maximum CP_{max}, where CP_{max} is the maximum *CP* combination that can be assigned for Zone 1, and CP_{min} is the minimum *CP* that can be assigned to the next zones (where all $CP <= CP_{min}$).

For every zone *z*, we will assign $CP(z) = CP_i$ from Equation (20), which can be used to map between zones and (N_0,N_1) combinations (TABLE):

$$i = \left\lceil z \div \left\lceil \frac{Count\ of\ all\ Zones}{Count\ of\ CPs\ between\ CP_{min} and\ CP_{max})} \right\rceil \right\rceil \qquad (25)$$

$$CP_1 = \left\lceil 1/\left\lceil \frac{6}{4} \right\rceil \right\rceil = \left\lceil \frac{1}{2} \right\rceil = 1$$

$$CP_2 = \left\lceil 2/\left\lceil \frac{6}{4} \right\rceil \right\rceil = \left\lceil \frac{2}{2} \right\rceil = 1$$

$$CP_3 = \left\lceil 3/\left\lceil \frac{6}{4} \right\rceil \right\rceil = \left\lceil \frac{3}{2} \right\rceil = 2$$

$$CP_4 = \left\lceil 4/\left\lceil \frac{6}{4} \right\rceil \right\rceil = \left\lceil \frac{4}{2} \right\rceil = 2$$

$$CP_5 = \left\lceil 5/\left\lceil \frac{6}{4} \right\rceil \right\rceil = \left\lceil \frac{5}{2} \right\rceil = 3$$

$$CP_6 = \left\lceil 6/\left\lceil \frac{6}{4} \right\rceil \right\rceil = \left\lceil \frac{6}{2} \right\rceil = 3$$

TABLE 11

Zones to (N_0,N_1) Cominations Mapping

Zone	CP	N_0	N_1
1	0.82	0	4
2	0.82	0	4
3	0.73	1	5
4	0.73	1	5
5	0.61	1	4
6	0.61	1	4

In this scheme we find that (N_0,N_1) will be dynamically selected by the nodes, depending on the path length, and the entire procedure described above will require minimal extra processing at the node, since the whole process is part of the network discovery phase.

5.4 Final Decision:

As in the hybrid scheme we will have K number of nodes, those nodes will collect T, and the node result will be 0 if the number of 1's are less than N_0, in other words if number of 0's collected are greater than or equal to T-N_0, and will send 1 if the number of 1's are greater than or equal to N_1. Otherwise all the collected data is sent (Equation 21).

$$Result_i = \begin{cases} 1, & n_i \geq N_1 \\ n_i, & N_0 < n_i < N_1 \\ 0, & n_i \leq N_0 \end{cases} \tag{26}$$

Out of K sensor nodes, t nodes send 1's, s nodes send 0's and $k - s - t$ nodes send all their observation so total send data Ω will be: $\Omega = \{1, \dots, 1; n_1, \dots, n_{k-s-t}; 0, \dots, 0;\}$

Our final decision, based on Bayes' Rule, is $H' = H1$ if $P[H_1|\Omega] \geq P[H_0|\Omega]$.

From the above rule we can derive the following relations:

$$\frac{P[\Omega|H_1]}{P[\Omega|H_0]} \geq \frac{1-p}{p} \tag{27}$$

where we can calculate $P[\Omega|H_1]$ from Equation (23), and $P[b_i = 0|H_\alpha]$, $P[b_i = 1|H_\alpha]$ and $P[b_i = 1|H_\alpha]$ from Equations (24), (25) and (26):

$$P[\Omega|H_\alpha] = \binom{k}{s}\binom{k-s}{t} P[b = 0|H_\alpha]^s \times P[b = 1|H_\alpha]^t \times P[n_i|H_\alpha]^{k-s-t} \tag{28}$$

$$P[b_i = 0|H_\alpha] = \sum_{n=T-N0_i}^{T} \binom{n}{T - N0_i - 1} (1 - p_\alpha)^{T-N_{0i}} p_\alpha^{i-(T-N_{0i})} \tag{29}$$

$$P[b_i = 1|H_\alpha] = \sum_{n=N1_i}^{T} \binom{n-1}{N_{1i} - 1} (1 - p_\alpha)^{n-N_{1i}} p_\alpha^{N_{1i}} \tag{30}$$

$$P[n_i|H_\alpha] = \binom{T}{n_i} (1 - p_\alpha)^{1-n_i} p_\alpha^{n_i} \tag{31}$$

CHAPTER 6

SIMULATION RESULTS

6.1 Enhanced Hybrid Detection Scheme

We simulated building WSN network 1000 time for each different number of node *K* is varied from 5 to 45 to get distribution of nodes and average path.

Fig. 34 shows the average path length increases as long with node count.

TABLE 12
Zones Usage Percantage

Percent/ Hub #	5	10	15	20	25	30	35	40	45
1	20.00%	10.00%	6.67%	5.00%	4.00%	3.33%	2.86%	2.50%	2.22%
2	58.00%	49.05%	43.31%	40.37%	37.19%	35.38%	34.27%	32.54%	32.02%
3	18.62%	27.81%	30.71%	32.17%	33.28%	33.94%	34.27%	35.26%	35.34%
4	3.28%	10.50%	14.51%	16.63%	18.40%	20.14%	21.22%	22.24%	22.77%
5	0.10%	2.38%	4.13%	4.98%	6.11%	6.31%	6.48%	6.88%	6.99%
6	0.00%	0.23%	0.63%	0.76%	0.91%	0.80%	0.83%	0.57%	0.65%
7	0.00%	0.03%	0.05%	0.09%	0.10%	0.08%	0.06%	0.02%	0.01%
8	0.00%	0.00%	0.01%	0.02%	0.02%	0.01%	0.01%	0.00%	0.00%
9	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%

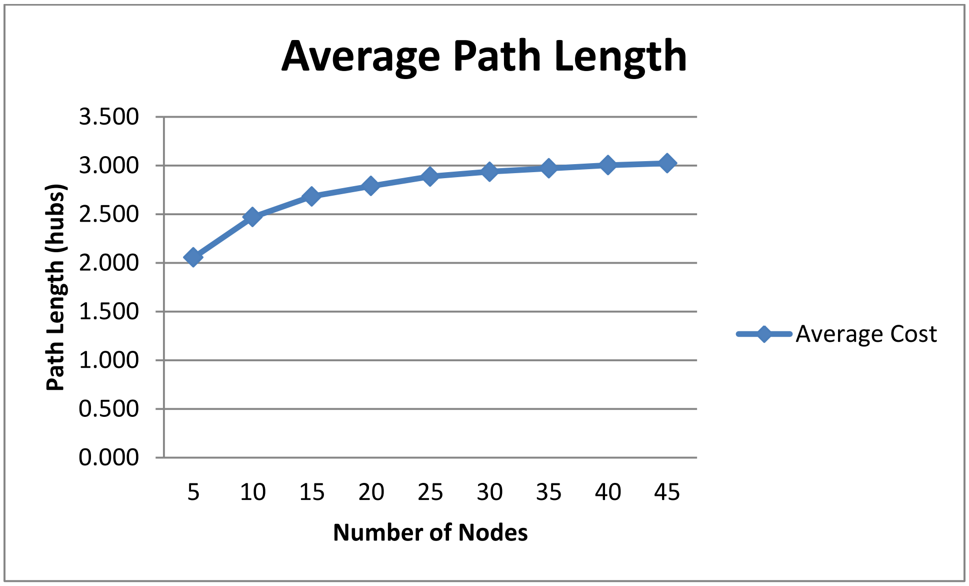

Fig. 34. Nodes Average Path Length

6.2 Power Consumption:

By applying "Telos" power model and using Equations (19), (20) and (21), Power calculation will be as following: Processing Power = $15 * 10^{-6}$ mA/sample, Rx power = $92 * 10^{-6}$ mA/bit, Tx Power = $84 * 10^{-6}$ mA/bit and T=5.

Centralized:

In centralized scheme the most power consumption process is the RF transmission, as can be seen in Fig. 35.

TABLE 13
Centrlized Scheme Power Consumption

Power\ Nodes	5	10	15	20	25	30
Er	0.002426	0.006763	0.011608	0.016461	0.021691	0.0267
Et	0.002215	0.006175	0.010598	0.015029	0.019805	0.024378
Ep	7.91E-05	0.000221	0.000379	0.000537	0.000707	0.000871
Total	0.00472	0.013158	0.022584	0.032027	0.042204	0.051949

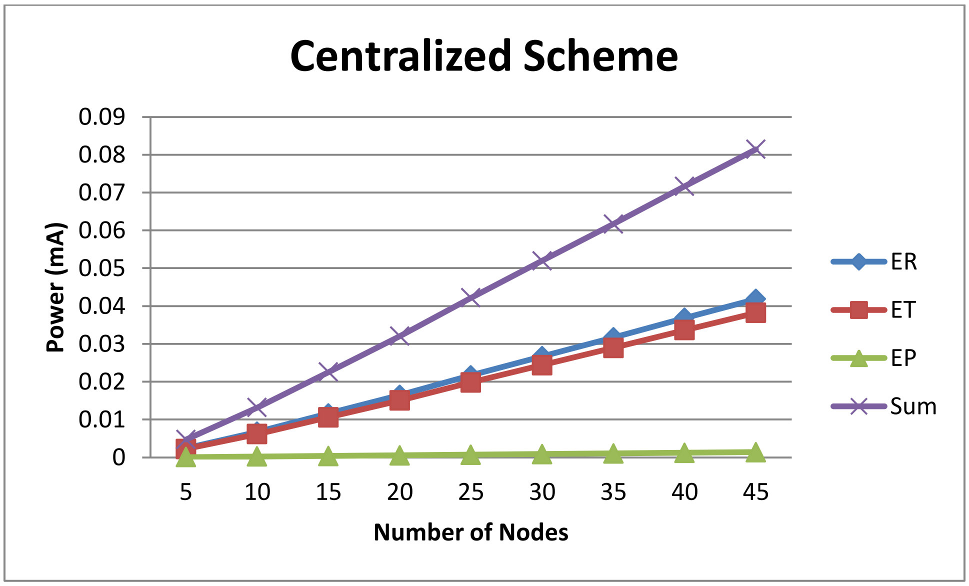

Fig. 35. Centrlized Scheme Power Consumption

Distributed:

Similar to centralized scheme the most power consumption process in the distributed is the RF transmission, as shown in Fig. 36.

TABLE 14
Distributed Scheme Power Consumption

Power\ Nodes	5	10	15	20	25	30
Er	0.000485	0.001353	0.002322	0.003292	0.004338	0.00534
Et	0.000443	0.001235	0.00212	0.003006	0.003961	0.004876
Ep	0.000396	0.001103	0.001893	0.002684	0.003537	0.004353
Total	0.001324	0.00369	0.006334	0.008982	0.011836	0.014569

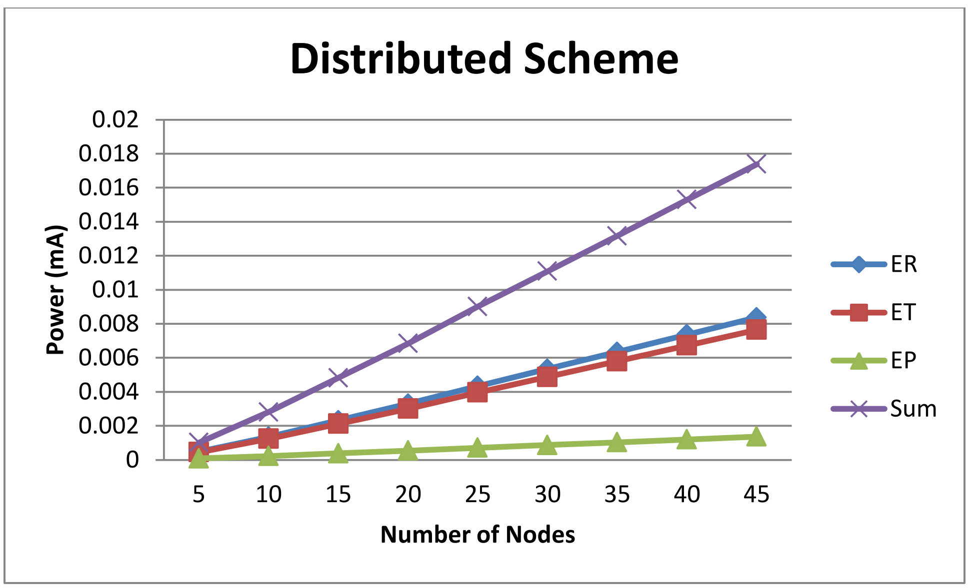

Fig. 36. Distributed Scheme Power Consumption

Hybrid (N0=1, N1=4):

We assume the following: $p = 0.5$, $p0 = 0.2$, $p1 = 0.7$, $T = 5$; K is varied from 10 to 30 (Fig. 37);

TABLE 15
Hybrid Scheme Power Consumption

Power\ Nodes	5	10	15	20	25	30
Er	0.001674	0.004667	0.00801	0.011359	0.014968	0.018425
Et	0.001529	0.004261	0.007313	0.010371	0.013667	0.016822
Ep	0.000202	0.000562	0.000965	0.001369	0.001804	0.00222
Total	0.003404	0.00949	0.016288	0.023098	0.030438	0.037467

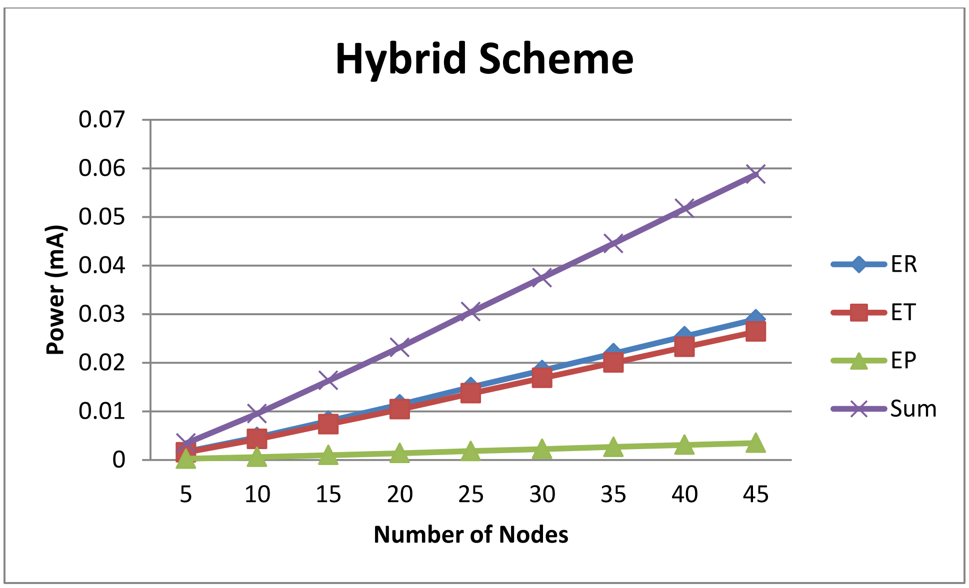

Fig. 37. Hybrid Scheme Power Consumption

Enhanced Hybrid:

We define the probability of node behavior as centralized *CP*; we get *CP* values out of possible combination of N_0 and N_1.

Fig. 38 shows the power consumption of enhanced hybrid scheme, where we assign $CP_{min}=0.25$ and $CP_{max}=0.70$.

TABLE 16
All (N_0,N_1) Combinations

N_0	N_1	CP
0	1	0.00
1	2	0.00
2	3	0.00
3	4	0.00
4	5	0.00
3	5	0.11
0	2	0.21
2	4	0.28
1	3	0.34
2	5	0.39
0	3	0.54
1	4	0.61
1	5	0.73
0	4	0.82
0	5	0.93

TABLE 17
Enhanced Hybrid Scheme Power Consumption

Power\ Nodes	5	10	15	20	25	30
Er	0.001584	0.004151	0.006919	0.009692	0.012621	0.015469
Et	0.001446	0.00379	0.006317	0.008849	0.011523	0.014124
Ep	0.000216	0.000646	0.001143	0.00164	0.002186	0.002702
Total	0.003246	0.008588	0.01438	0.020181	0.02633	0.032295

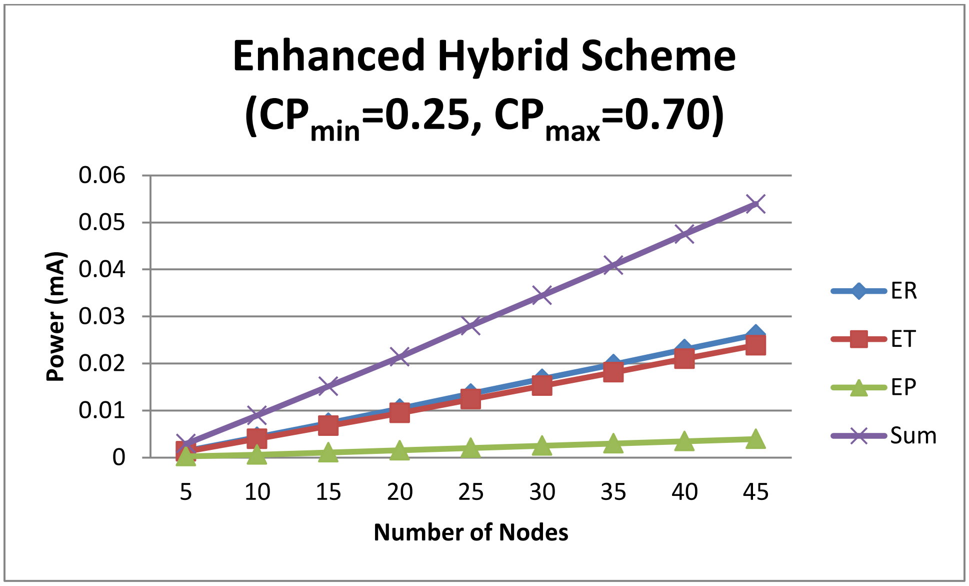

Fig. 38. Enhanced Hybrid Scheme Power Consumption

All Energy Schemes:

TABLE 18
Power Consumption of all Schemes

Power\Nodes	5	10	15	20	25	30
Centralized	0.00472	0.013158	0.022584	0.032027	0.042204	0.051949
Distributed	0.001324	0.00369	0.006334	0.008982	0.011836	0.014569
Hybrid	0.003404	0.00949	0.016288	0.023098	0.030438	0.037467
Advanced	0.003246	0.008588	0.01438	0.020181	0.02633	0.032295

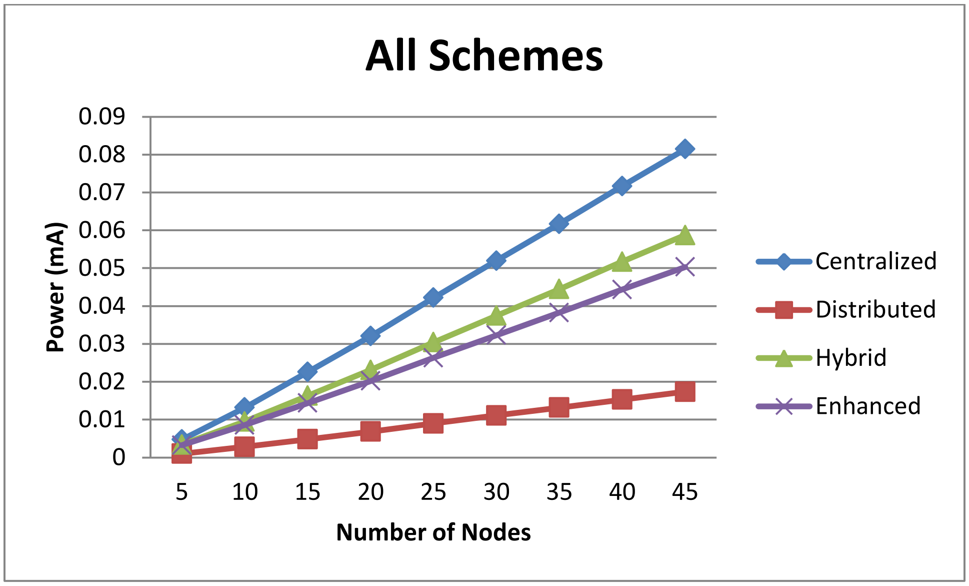

Fig. 39. Power Consumption of all Schemes

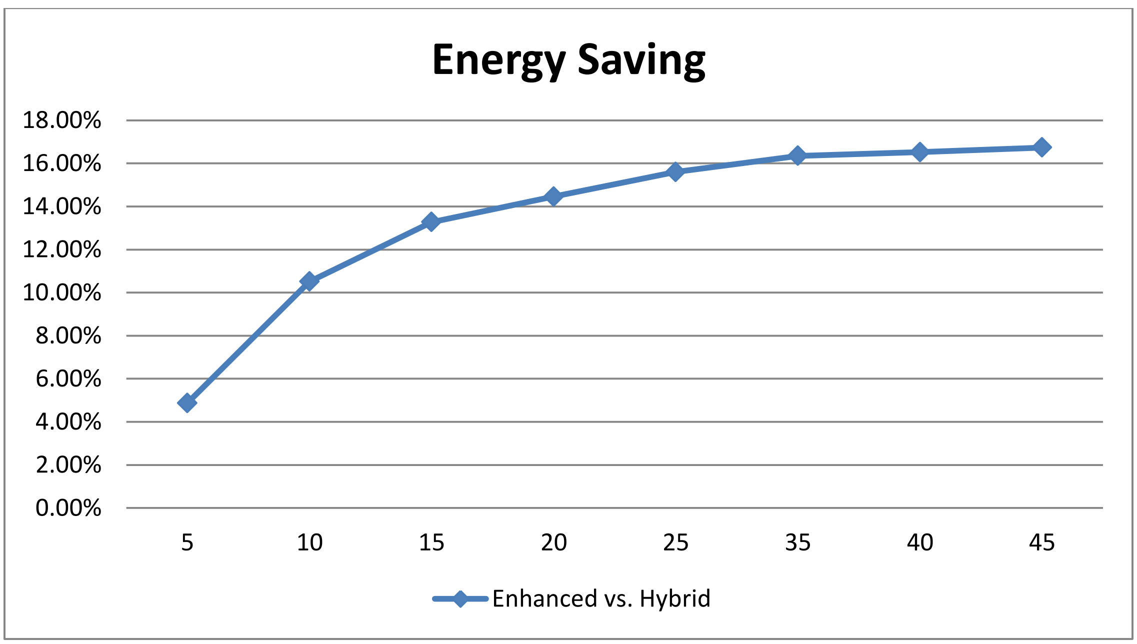

Fig. 40. Enhanced vs. Hybrid Power Consumption

6.3 Accuracy behavior:

In our simulation we are able to get the same result which is given in [46], where we used in hybrid (N_0,N_1) = (1,3), (1,4) and (0,4) as shown in

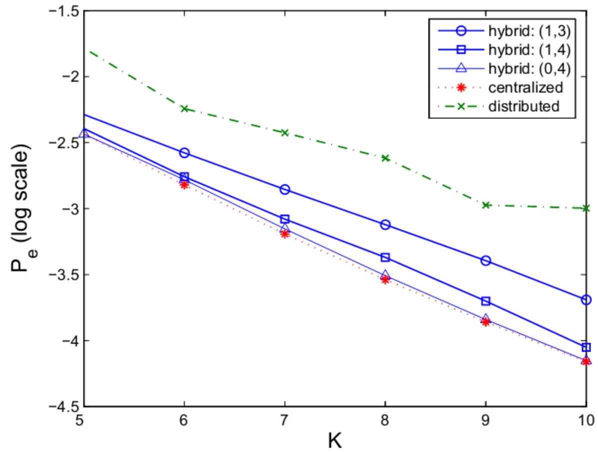

Fig. 41. Comparison of Three Schemes in Detection Accuracy [46]

Centralized Scheme Pe:

TABLE 19
Centrlized Scheme Propability of Error

Nodes	10	15	20	25	30	35
Pe	6.93E-05	1.47E-06	3.29E-08	7.58E-10	1.78E-11	4.24E-13
Log10(Pe)	-4.15935	-5.83216	-7.48262	-9.12031	-10.7497	-12.3731

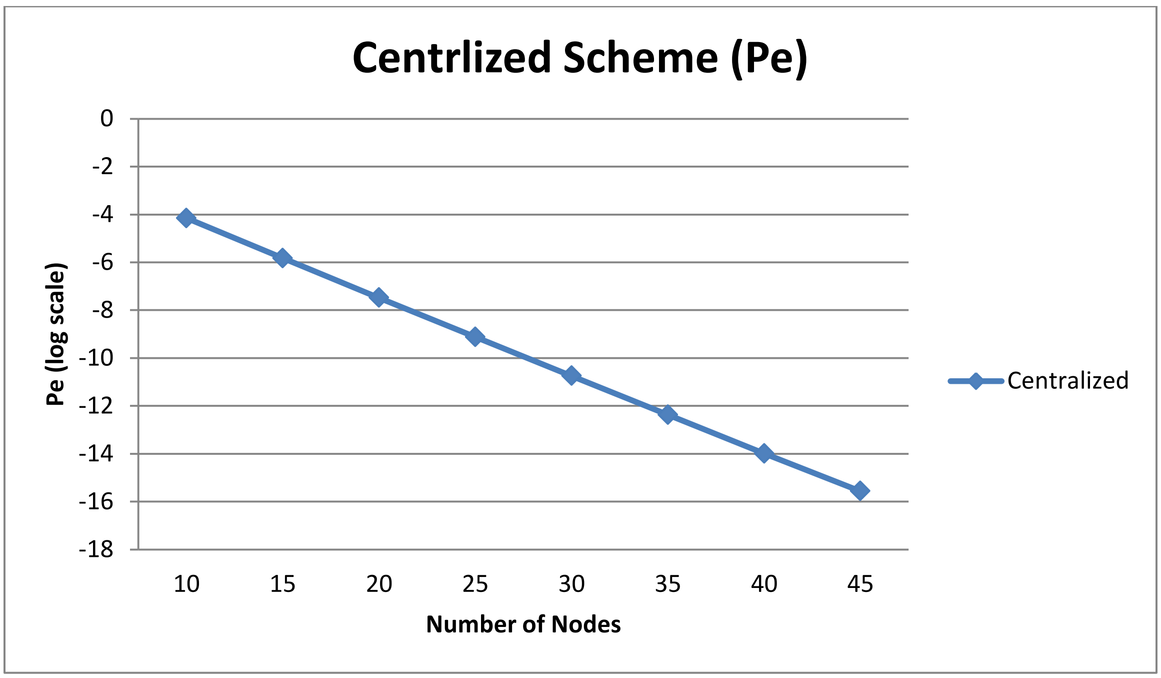

Fig. 42. Centrlized Scheme Propability of Error

Distributed Scheme Pe:

TABLE 20
Distributed Scheme Propability of Error

Nodes	10	15	20	25	30	35
Pe	1.01E-03	6.89E-05	4.95E-06	3.67E-07	2.77E-08	2.12E-09
Log10(Pe)	-2.99723	-4.16209	-5.30519	-6.43572	-7.55793	-8.67419

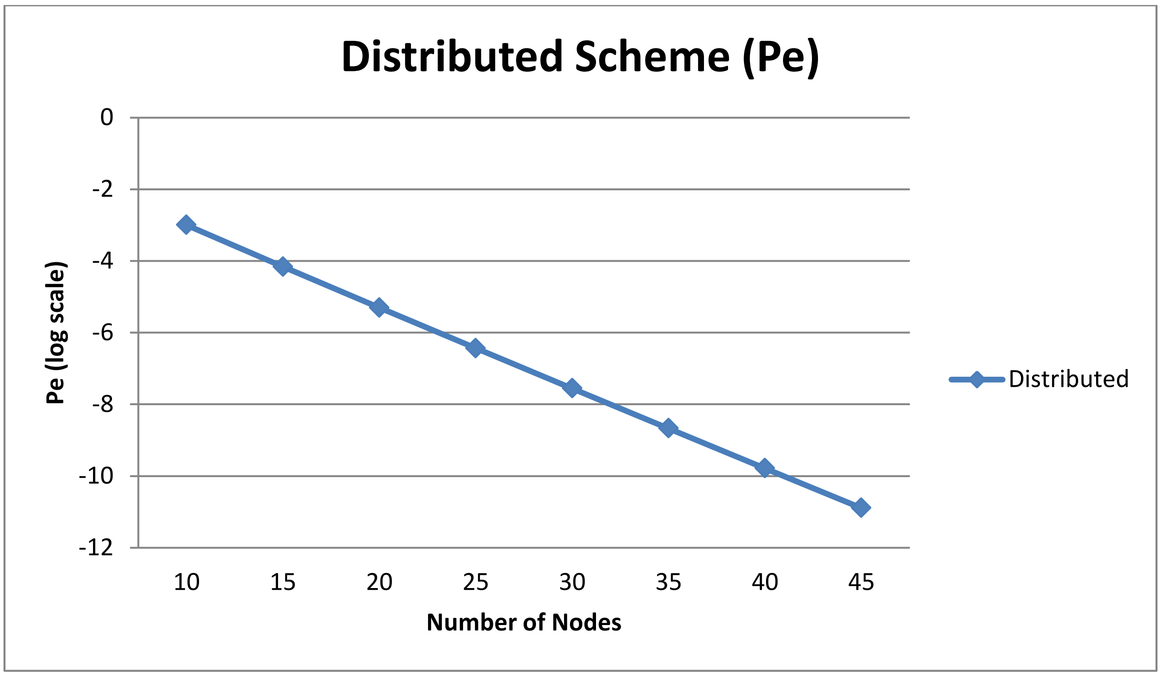

Fig. 43. Distributed Scheme Propability of Error

Hybrid Scheme Pe:

TABLE 21
Hybrid Scheme Propability of Error

Nodes	10	15	20	25	30	35
Pe	1.17E-04	6.67E-06	4.55E-07	3.27E-08	8.07E-10	6.70E-11
Log10(Pe)	-3.93212	-5.17579	-6.34189	-7.48505	-9.09319	-10.1742

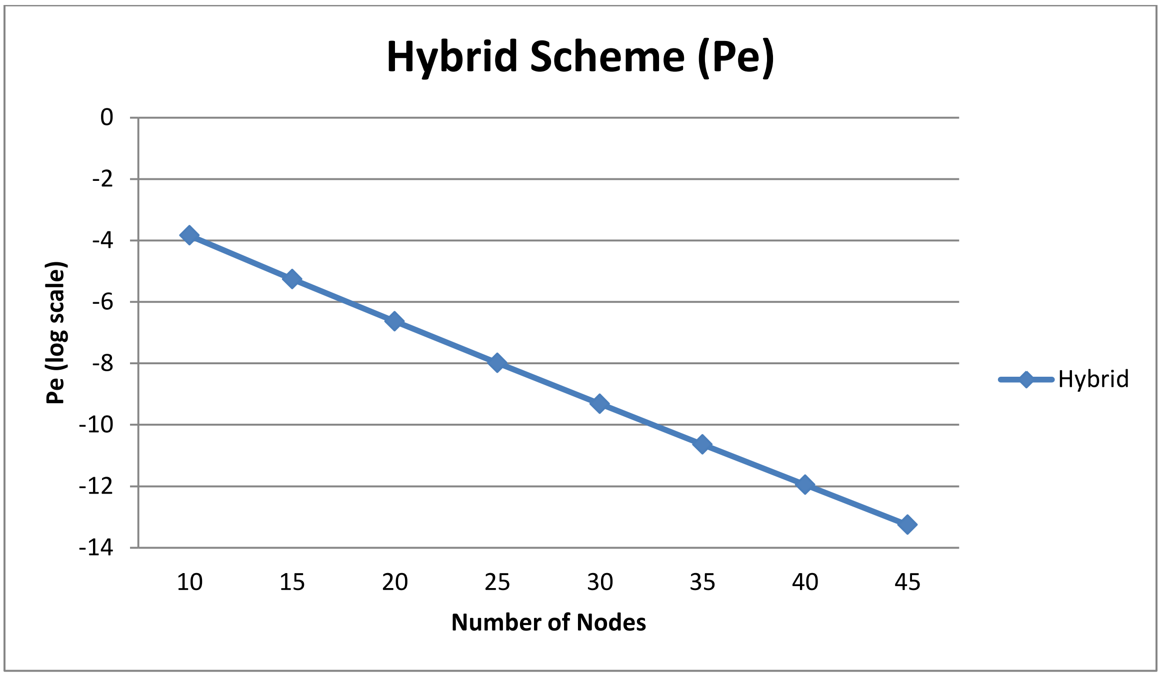

Fig. 44. Hybrid Scheme Propability of Error

Enhanced Hybrid Scheme Pe:

TABLE 22
Enhaced Scheme Propability of Error

Nodes	10	15	20	25	30	35
Pe	1.17E-04	6.67E-06	4.55E-07	3.27E-08	8.07E-10	6.70E-11
Log10(Pe)	-3.93212	-5.17579	-6.34189	-7.48505	-9.09319	-10.1742

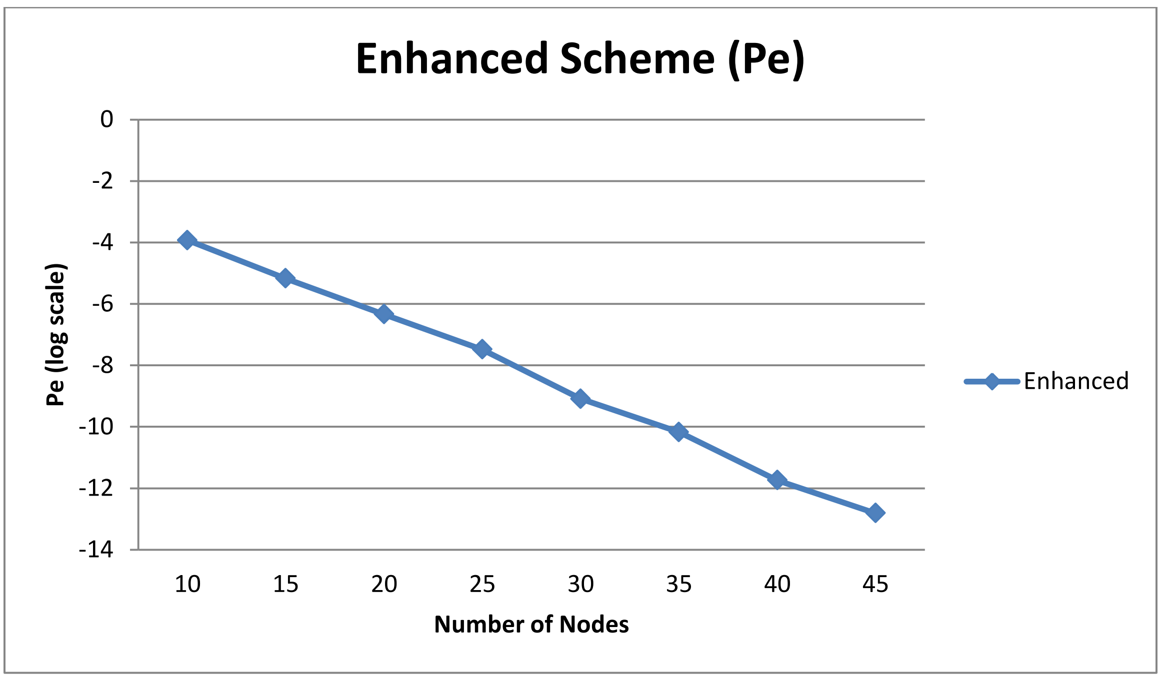

Fig. 45. Enhaced Scheme Propability of Error

All Schemes:

In comparison to hybrid scheme, in our scheme gives a very close level of accuracy, as illustrated in Fig. 46.

TABLE 23
Propability of Error for all Schemes

Nodes	10	15	20	25	30	35
Centralized	-4.15935	-5.83216	-7.48262	-9.12031	-10.7497	-12.3731
Distributed	-2.99723	-4.16209	-5.30519	-6.43572	-7.55793	-8.67419
Hybrid	-3.84127	-5.26152	-6.63689	-7.98678	-9.32022	-10.6421
Advanced	-3.93212	-5.17579	-6.34189	-7.48505	-9.09319	-10.1742

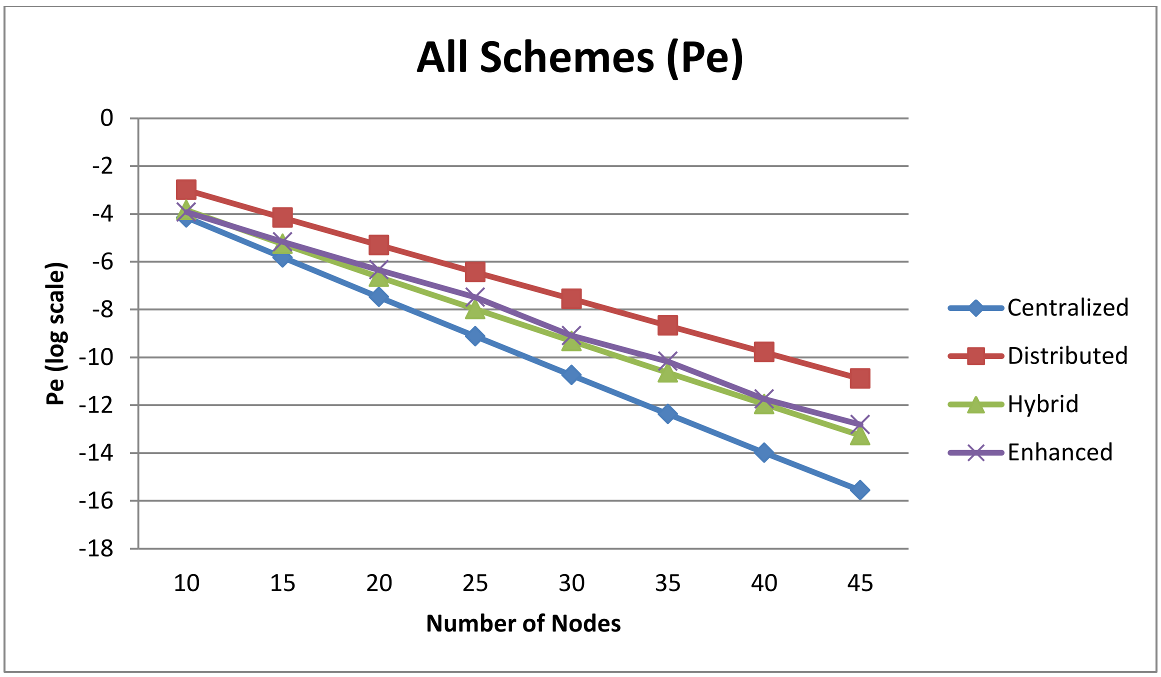

Fig. 46. Propability of Error for all Schemes

CHAPTER 7

CONCLUSION

The primary purpose of this thesis is to enhance the event collection and detection capability of current WSN schemes. Throughout this thesis the focus has been on three available detection schemes. The criteria for comparing my proposed scheme with other schemes have been mainly accuracy and energy consumption.

Two of the available schemes – centralized and distributed schemes - are basic and have no flexibility. The third scheme – the hybrid scheme– switches between the two previous schemes to balance accuracy and energy; nevertheless all the nodes remain with fixed configuration and limited flexibility.

In contrast to earlier schemes, the proposed scheme enhances the hybrid schemes and is designed to be more flexible in order to balance the power consumption and the detection accuracy at the node level. Every node is flexible in deciding how to behave, i.e. whether to be more centralized or distributed.

To be able to compare the tradeoff between accuracy and power consumption in these schemes the realistic model was used. This is one of the most significant weaknesses of the previous research, in which all power consumption calculations have been based solely on a hypothetical model. In the proposed scheme this weakness has been overcome by using a real WSN model, which produces realistic results.

As can be seen from the simulation results, our scheme saves a substantial amount of energy compared to the hybrid scheme, while it retains accuracy to almost the same degree. In addition, our scheme deals more efficiently with larger network area and denser node-number.

Although the focus of this thesis has been on reducing overall power consumption, other future enhancements are also possible, for example enhancing overall network lifetime by improving (N_0, N_1) selection techniques where the following factors can be considered: individual node power level, event occurrence probability at each node, next-hop distance and real Tx, Rx power.

CHAPTER 8

REFERENCES

[1] Ittipong Khemapech, Alan Miller, and Ishbel Duncan, "Simulating Wireless Sensor Networks," University of St Andrews, St Andrews, 2005.

[2] Waltenegus Dargie and Christian Poellabauer, *FUNDAMENTALS OF WIRELESS SENSOR NETWORKS, THEORY AND PRACTICE*. UK: A John Wiley and Sons, Ltd., Publication, 2010.

[3] Mawan Ihsan Shukur, Lee Sheng Chyan, and Vooi Voon Ya, "Wireless Sensor Networks:Delay Guarentee and Energy Efficient MAC Protocols," *World Academy of Science, Engineering and Technology*, vol. 50, no. 2009, 2009.

[4] Thomas Haenselmann, *Sensornetworks*., 2006.

[5] Jon S. Wilson, Ed., *Sensor Technology Handbook*. USA: Newnes, 2005.

[6] Chris Townsend and Steven Arms. (2004) Wireless Sensor Networks: Principles and Applications.

[7] Wikipedia. (2011) Wireless sensor network. [Online]. http://en.wikipedia.org/wiki/Wireless_sensor_network

[8] Adi Kumar, D Janakiram, and G Ashok Kumar, "Operating Systems for Wireless Sensor Networks: A Survey," Indian Institute of Technology, Madras, Technical Report 2007.

[9] Ian F. Akyildiz and Mehmet Can Vuran, *Wireless Sensor Networks*.: John Wiley & Sons Ltd., 2010.

[10] Crossbow Technology. [Online]. http://www.xbow.com

[11] Cricket indoor location system. [Online]. http://nms.lcs.mit.edu/projects/cricket

[12] SunSPOT mote specifications. [Online]. http://www.sunspotworld.com

[13] I.F. Akyildiz, W. Sui, Y. Sankarasubramaniam, and E. Cayirc, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, pp. 393–422, 2002.

[14] Holger Karl and Andreas Willig, *Protocols and Architectures for Wireless Sensor Networks*.: John Wiley & Sons, Ltd., 2005.

[15] Edgar H. Callaway, *Wireless Sensor Networks – Architectures and Protocols*.: AUERBACH PUBLICATIONS, 2004.

[16] Rajesh Yadav, Shirshu Varma, and N. Malaviya, "A SURVEY OF MAC PROTOCOLS FOR WIRELESS SENSOR," *UbiCC Journal*, vol. 4, no. 3, pp. 827-833, August 2009.

[17] Wei Ye, John Heidemann, and Deborah Estrin, "An Energy-Efficient MAC protocol for Wireless," *IEEE INFOCOM*, vol. 2, pp. 1567-1576, June 2002.

[18] Miguel A. Erazo and Yi Qian, "An Energy-Efficient MAC Protocol for Wireless Sensor Networks for Wide Area Large Scale Environmental Monitoring," in *Fourth*

LACCEI International Latin American and Caribbean Conference for Engineering and Technology, Mayagüez, 2006.

[19] G. Lu, B. Krishnamachari, and C.S. Raghavendra, "An adaptive energy-efficient and low-latency MAC for data gathering in wireless sensor networks," in *Parallel and Distributed Processing Symposium*, 2004, p. 224.

[20] Jamal N. Al-Karaki and Ahmed E. Kamal, "Routing techniques in wireless sensor networks: a survey," *Wireless Communications, IEEE*, vol. 11, no. 6, pp. 6-28, December 2004.

[21] Subhas Chandra Mukhopadhyay and Henry Leung, Eds., *Advances in Wireless Sensors and Sensor Networks*, 1st ed.: Springer, 2010.

[22] OpenWSN. [Online]. http://openwsn.berkeley.edu/

[23] ZigBee Alliance. (2008, January) ZIGBEE SPECIFICATION. Document.

[24] (2011) ZigBee Alliance. Zigbee Specification. [Online]. http://www.zigbee.org/

[25] (2007, May) WirelessHART Data Sheet. Document.

[26] (2010, March) WirelessHART Training Resources. [Online]. http://www.hartcomm.org/protocol/training/training_resources_wihart.html

[27] N. Kushalnagar, G. Montenegro, and C. Schumacher. (2007, August) IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals. RFC. [Online]. http://datatracker.ietf.org/doc/rfc4919/?include_text=1

[28] ISA. (2011) ISA100 Wireless Compliance Institute. [Online].

http://www.isa100wci.org

[29] Wei Dong, Chun Chen, Xue Liu, and Jiajun Bu, "Providing OS Support for Wireless Sensor Networks: Challenges and Approaches," *Communications Surveys & Tutorials, IEEE*, vol. 12, no. 4, pp. 519-530, May 2010.

[30] (2011) TinyOS Home Page. [Online]. http://www.tinyos.net/

[31] A. K. Dwivedi, M. K. Tiwari, and O. P. Vyas, "Operating Systems for Tiny Networked Sensors: A Survey," *International Journal of Recent Trends in Engineering*, vol. 1, no. 2, pp. 152-157, May 2009.

[32] Jens Kühner, *Expert.NET Micro Framework*, Dominic Shakeshaft, Ed.: Apress, 2008.

[33] Xiaodong Xian, Weiren Shi, and He Huang, "Comparison of OMNET++ and other simulator for WSN simulation," in *3rd IEEE Conference on Industrial Electronics and Applications, ICIEA*, Singapore, 2008, pp. 1439-1443.

[34] Miloš Jevtich, Nikola Zogovich, and Goran Dimich, "Evaluation of Wireless Sensor Network Simulators," in *17th Telecommunication forum TELFOR*, Belgrade, 2009.

[35] E. Egea-López, J. Vales-Alonso, A. S. Martínez-Sala, P. Pavón-Mariño, and J. García-Haro, "Simulation Tools for Wireless Sensor Networks," in *Summer Simulation Multiconference - SPECTS*, 2005.

[36] (2011) ns-2: The Network Simulator. [Online]. http://nsnam.isi.edu/nsnam/index.php

[37] Ahmed Sobeih et al., "J-Sim: A Simulation Environment for Wireless Sensor Networks," in *The 38th Annual Simulation Symposium (ANSS'05)*, 2005.

[38] Paulo Neves, Iúri Veiga, and Joel Rodrigues, "G-JSIM — a GUI tool for Wireless Sensor Networks simulations under J-SIM," in *IEEE International Symposium on Consumer Electronics, 2008. ISCE 2008.*, 2008, pp. 1 - 4.

[39] (2011) Castalia: A simulator for WSNs. [Online]. http://castalia.npc.nicta.com.au/

[40] A. Rastegarnia and V. Solouk, "Performance evaluation of Castalia Wireless Sensor Network simulator," in *34th International Conference on Telecommunications and Signal Processing (TSP)*, 2011.

[41] (2011) MiXiM project. [Online]. http://mixim.sourceforge.net

[42] M. Imran, A.M. Said, and H. Hasbullah, "A survey of simulators, emulators and testbeds for wireless sensor networks," in *International Symposium in Information Technology (ITSim)*, 2010, pp. 897 - 902.

[43] Li Xiangqun et al., "OMNeT++ and Mixim-based protocol simulator for satellite network," in *Aerospace Conference*, 2011, pp. 1 - 9.

[44] D. Weber, J. Glaser, and S. Mahlknecht, "Discrete Event Simulation Framework for Power Aware Wireless Sensor Networks," in *5th IEEE International Conference on Industrial Informatics*, 2007, pp. 335 - 340.

[45] Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers, and Keying Ye, *Probability and Statistics for Engineers and Scientists*, 9th ed., Sally Yagun, Ed.: Pearson Education International, 2011.

[46] Lige Yu, Lin Yuan, Gang Qu, and Anthony Ephremides. Energy-Driven Detection Scheme with Guaranteed Accuracy.

[47] (2010) MEMSIC: Wireless Sensor Networks Modules. [Online]. http://www.memsic.com/products/wireless-sensor-networks/wireless-modules.html

[48] Joseph Polastre, Robert Szewczyk, and David Culler, "Telos: Enabling Ultra-Low Power Wireless Research," in *IPSN 2005. Fourth International Symposium on Information Processing in Sensor Networks*, 2005, pp. 364 - 369.

[49] (2003) TelosB Mote TPR2420: Datasheet. Datasheet.

[50] (2004, May) Telos: PRELIMINARY Datasheet.

[51] (2004) MSP430: Datasheet.

[52] Ruixin Niu, Michael Moore, and Dale Klamer, "Decision Fusion in a Wireless Sensor Network with a Large Number of Sensors," in *7th IEEE Conference on Information Fusion (ICIF 2004)*, 2004.

[53] Wikipedia. (2011) Sensor Node. [Online]. http://en.wikipedia.org/wiki/Sensor_node

[54] *Communications Engineering Desk Reference*. Oxford, UK: Elsevier Inc, 2009.

[55] Charlie Stiernberg. (2010, October) Five Things Every Scientist and Engineer Should Know about Wireless Sensor Measurements. [Online]. http://www.laboratoryequipment.com/article-wireless-sensor-measurements-0903.aspx

[56] Jennifer Yick, Biswanath Mukherjee, and Dipak Ghosal, "Wireless sensor network survey," *Computer Networks*, vol. 52, pp. 2292–2330, April 2008.

[57] Lige Yu and Anthony Ephremides, "Detection performance and energy efficiency trade-off in a sensor network," , 2003.

[58] Wikipedia. (2010, may) Wikipedia. [Online].
http://en.wikipedia.org/wiki/Wireless_sensor_networks

[59] Yu-Chee Tseng. (2007) Information processing and data management in wireless sensor networks.

[60] Francesco Marcelloni and Massimo Vecchio, "A Simple Algorithm for Data Compression in Wireless Sensor Networks," vol. 12, no. 6, 2008.

[61] Xuanwen Luo, Ming Dong, and Yinlun Huang, "On Distributed Fault-Tolerant Detection in Wireless Sensor Networks," vol. 55, no. 1.

[62] Daniel Klan, Katja Hose, and Kai-Uwe Sattler. Developing and Deploying Sensor Network Applications with AnduIN.

[63] Jayant Gupchup, Randal Burns, and Andreas Terzis. (2007) Model-Based Event Detection in Wireless Sensor Networks.

[64] An Energy Efficient Adaptive Event Detection Scheme for Wireless Sensor Network.

[65] Vladimir Bychkovskiy, Seapahn Megerian, Deborah Estrin and Miodrag Potkonjak. A Collaborative Approach to In-Place Sensor Calibration.

VITA

Personal Information:

- Name: Mohammad Nabeel Mousa Al-Sallout
- Nationality: Palestinian
- Birth: December 1983, Saudi Arabia
- Social Status: Married with 1 kid
- Current Address: Saudi Arabia - Al-Khobar
- Permanent Address: Saudi Arabia – Medina Monawarah
- Email: mhd2111@gmail.com
- Phone No.: 00966-500189750

Education:

- 2012: *Master* of Science in Computer Engineering KFUPM
- 2006: *Bachelor* of Science in Computer Engineering KFUPM

Work Experience:

- Currently: Mobily, Public WiFi Network Project, Saudi Arabia.
- 2011-2012: ZTE – Zain, USF GSM/UMTS Network Project, Saudi Arabia.
- 2009-2011: SAMSUNG-Mobily, WiMax Network Project, Saudi Arabia.

- 2007-2009: Alyaum Media, Microsoft Dynamics Project, Saudi Arabia.
- 2006-2007: Eastern Province Emirate, HR System Project, Saudi Arabia.