Contents

Chapter 6

Chapter 7

Abstract

According to MIT technology review, Wireless Sensor Networks (WSN) is one of ten technologies that will change the world. This technology has been able to transform computing, medicine, transportation, and manufacturing to facilitate our daily life. Therefore, WSN attracted the attention of many researchers around the world and it becomes the heart of many applications such as surveillance, machine failure diagnosis, weather forecast, intelligent environments, intelligent campuses, and chemical/biological detection. In addition, WSNs are used in many critical applications that require reliable and accurate data as well as long lifetime due to the deployment environment.

Nonetheless, WSN still suffers from many limitations such as the computing capabilities, and the allocated bandwidth. However, the limited energy source is the dominant factor where energy starvation occurs due to the large number of messages need to be transferred through the network. This thesis proposes several techniques including predictors and fuzzy logic for minimizing the data sent by different nodes to the base station in order to maximize the WSN lifetime.

However, such techniques should not be confused with traditional methods of data reduction such as clustering, grouping, and principle component analysis. At the same time, the proposed approaches are used for Reliable and Efficient Data Reduction (REDR) in WSN. The thesis also investigates the data reduction in single and multimodal

WSN. In addition, it utilizes the concept of exponential smoothing predictors including single and double exponential algorithms as well as the fuzzy logic to provide data reduction and reliability.

Moreover, part of our work is dedicated for deploying of WSN inside greenhouses for controlling the environmental conditions. These greenhouses are used extensively by botanists, commercial plant growers, and dedicated gardeners. Particularly in cool climates, greenhouses are useful for growing and propagating plants because they both allow sunlight to enter and prevent heat from escaping consequently. They provide the plants with the suitable environmental conditions such as temperature, humidity. The transparent covering of the greenhouse allows the light to enter unhindered, where it warms the interior as it is absorbed by the material within. In addition, greenhouses may be used in the winter to increase the temperature by making some modifications on the covering of the greenhouse.

In this thesis, the importance of Wireless Sensor Networks (WSNs) in Greenhouses applications is demonstrated by using single WSN prototype for different applications. In addition, the proposed REDR energy saving technique is utilized to increase the wireless sensor node lifetime inside the Greenhouse. Moreover, a simple Fuzzy Logic Controller (FLC) is used in taking right decisions and avoiding false alarms. Nevertheless, off-shelf components and none-expensive devices are showed to be effectively to be used for greenhouse applications through Internet based monitoring.

For the purpose of comparison, threshold and tolerance based algorithms are implemented. Through large number of experiments, the proposed approaches are tested using real data and WSN prototypes as well as through simulation.

List of Abbreviations

ADC	Analog-to-Digital Converter
ANFIS	Adaptive Neuro Fuzzy Inference System
AOA	Angle of Arrival
ASCENT	Adaptive Self-Configuring Sensor Networks Topologies
ASIC	Application Specific Integrated Circuits
CDMA	Code Division Multiple Access
CMOS	Complementary Metal Oxide Semiconductor
CO_2	Carbon Dioxide
CRC	Cyclic Redundancy Checksum
CSMA	Carrier Sense Multiple Access
CV	Coefficient of Variation
CZ	Comfort Zone
DES	Double Exponential Smoothing
DMS	Dynamic Modulation Scaling
DPM	Dynamic Probabilistic Model
DPS	Dual Prediction Scheme
DSAP	Directional Source-Aware Protocol
DSP	Digital Signal Processor
DTM	Dynamic Thermal Management
DVS	Dynamic Voltage Scaling
EEPROM	Electrically Erasable Programmable Read Only Memory
ES	Exponential Smoothing
FCC	Federal Communications Commission
FDMA	Frequency Division Multiple Access
FHSS	Frequency Hopping Spread Spectrum
FIS	Fuzzy Inference System
FLC	Fuzzy Logic Controller
FPGA	Field Programmable Gate Array
GA	Genetic Algorithm
GAF	Geographical Adaptive Fidelity

GeRaF	Geographic Random Forwarding
GFSK	Gaussian Frequency Shift Keying
GPS	Global Positioning System
HCV	Heating–Cooling Ventilating
IC	Integrated Circuit
IEEE	Institute of Electrical and Electronics Engineers
LEACH	Low-Energy Adaptive Clustering Hierarchy
LMS	Least Mean Square
MA	Moving Average
MAC	Medium Access Control
MEMS	Micro Electro Mechanical System
MISO	Multi Input Single Output
MPPT	Maximum Power Point Tracker
MSE	Mean Square Error
NI	National Instrument
NiCad	Nickel-Cadmium
Nimh	Nickel Metal Hydride
NiZn	Nickel-Zinc
ODE	Offset Delay Estimation
PAN	Personal Area Network
PCM	Pulse Code Modulation
PDA	Personal Digital Assistant
PDF	Probability Density Function
PHY	Physical Layer
PM	Power Management
PPS	Precise Positioning System
PREMON	PREdiction MONitoring
QoS	Quality of Service
RCR	Remote Clock Reading
REDR	Reliable and Efficient Data Reduction
RSS	Received Signal Strength
SD	Standard Deviation

SES	Single Exponential Smoothing
SMAC	Carrier Medium Access Control
SMP	Sensor Management Protocol
SNR	Signal to Noise Ratio
SPS	Standard Positioning System
SQDDP	Sensor Query and Data Dissemination Protocol
STEM	Sparse Topology and Energy Management
SVE	Set-Valued Estimation
T1FLC	Type 1 Fuzzy Logic Controller
T2FLC	Type 2 Fuzzy Logic Controller
TADAP	Task Assignment and Data Advertisement Protocol
TC	Topology Control
TDMA	Time Division Multiple Access
TDOA	Time Difference of Arrival
TES	Triple Exponential Smoothing
TOA	Time for Arrival
TTP	Time Transmission Protocol
ULPS	Ultra Low Power Systems
US	United States
UTC	Universal Coordinated Time
UWB	Ultra Wideband
VI	Virtual Instrument
WPAN	Wireless Personal Area Network
WSN	Wireless Sensor Networks

List of Figures

List of Tables

Chapter 1

Introduction

The need for a new technology model that collects and analyzes data from the surrounding environment has led to the emergence of Wireless Sensor Networks. As shown in Fig. 1.1, WSN is a network of wireless battery-powered sensors that monitor our environment, our machines, and even us. These sensors are self-contained units equipped with a radio transceiver, a small processor, and an energy source [91]. They are deployed throughout a monitored field, cooperate, establish a routing topology, and transmit data back to a collection point for automatic control or human evaluation. The capabilities of WSNs make them the first candidate for thousands of applications including environmental monitoring, military surveillance and reconnaissance, infrastructure health monitoring, smart homes, patient monitoring, and many other applications. It is reasonable to expect that in 10-15 years that the world will be covered with WSNs accessing to them via the Internet.

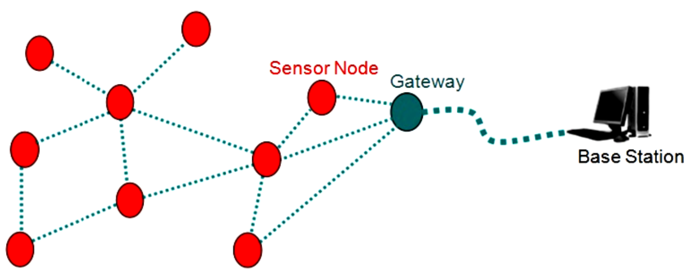


Fig. 1.1 WSN architecture

A sensor node is responsible for measuring certain quantity, creating packets, and sending these packets to a base station via multi hop transmission. However, it was found that the radio communication among the different nodes consumes the majority of the available energy. For instance, Verdone, et al stated that the energy cost of transmitting 1 Kb of information a distance of 100m is approximately the same as that of executing 3 million instructions [78]. Consequently, the radio messages are the main reason for shortening the node's lifetime. Many devices such as Mica2 and MicaZ that are used in WSN run on double AA batteries. Depending on the activity level of a node, its lifetime may run only for a few days if no power management schemes are used.

This thesis focuses on the sensed data not on the overhead data reduction since sensed data has the major effect on the lifetime of WSN. Two different types of WSNs are considered including single, and multi modal networks. In single modal networks, each sensor is assumed to measure a single feature from the sensed environment while in multimodal WSNs; a sensor may sense multiple features at the same time such as temperature, humidity, and pressure. Nowadays, new smart sensors are used to sense multiple features and report them in one message. These sensors help to provide fast and accurate readings to the monitoring environment and eliminate redundant hardware. For example, in Great Duck Island project [31], sensors have been deployed in fixed locations on the island trying to better understand the behavior of leach’s petrel by collecting data such as humidity, rainfall, and temperature on constant basis.

Quality of service (QoS) in WSN can be evaluated based on many parameters including accuracy, energy efficiency, reliability, latency, fault tolerance, scalability, bandwidth, and memory size. To ensure the performance of our proposed approach, the effect of applying some of the predictors on sensor nodes for these parameters will be investigated.

Accuracy is the difference between the real world values and the provided results. This implies that observations of most of the spatial phenomena are usually considered as estimates of the true value. On the other hand, precision is the probability of which a given accuracy is achieved. Fig. 1.2 shows the relation between the accuracy of the measurements, precision, and the expected lifetime.

The difference between observed and true (or accepted as being true) values, as mentioned, indicates the accuracy of the observations. The extent to which results are consistent over time and an accurate representation of the total population under study is referred to as reliability. Therefore, the data reliability in terms of its accuracy will be studied. Power efficiency is an important parameter since it determines the lifetime of nodes. It is expected that the higher the power efficiency the longer the lifetime.

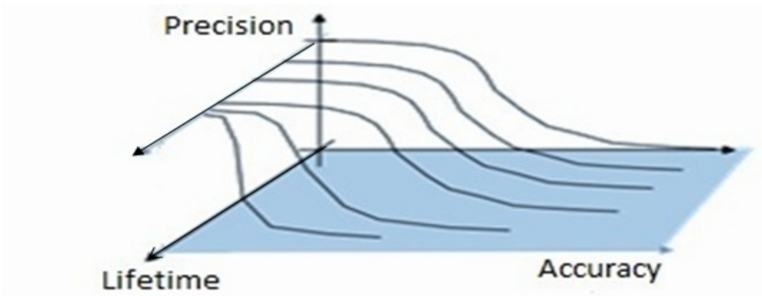


Fig. 1.2 Accuracy, precision, and the expected lifetime [33]

Scalability is another measurement of QoS in WSN where the network should be scalable and flexible to the enlargement of the network scale. Some of the used approaches to satisfy these parameters include clustering, multi-hop delivery, localization of computation, and data processing [38]. Finally, memory size limitations may lead to the degradation of the WSN performance since it stands against enhancing the WSN networking capabilities.

There are many trades off among these QoS parameters. For instance, the accuracy can be increased by transmitting more sensor readings, which in turn increasing the radio messages and reducing the power efficiency; consequently the lifetime is reduced. Therefore, designing a QoS model based on a specific application scenario allows us to identify key QoS metrics from which a feasible QoS management scheme potentially involving trade-offs can be derived.

1.1. Thesis's Objective

The main objective of this thesis is developing efficient and data reduction techniques in order to prolonging the overall lifetime of WSNs by minimizing the transmission among the sensor nodes. Additionally, a novel technique for detecting and reporting the plants diseases inside greenhouses is proposed.

This objective is achieved by employing different and more efficient approaches than those proposed in other research studies. These approaches and research steps are listed below:

1. Exponential Smoothing Statistical methods are employed as predictors to forecast the next sensors readings. This type of predictor is used in many applications such as inventory control applications, tracking, finance, and many other applications.
2. Fuzzy Logic algorithm is used in the multi-modal WSN to produce a certain decision at which the node has to send its information to the base station for evaluation. FLC is optimized through using the concept of Adaptive Neuro Fuzzy Inference System (ANFIS) [35].
3. The data reliability of the predicted values is investigated.

4. The performance of the proposed algorithms is examined using a real collected data available at [44]. In these experiments, a sample of intelligent environment WSN is used where the data was collected by indoor sensors at the Intel Berkley Research Lab.
5. Different types of WSNs including single modal and multi-modal networks are simulated based on different network topologies as well as communication ranges and sensing ranges. Such experiments simulate outdoor WSN suitable for critical applications such as battlefield and habitat monitoring.
6. The exponential smoothing predictors are examined on a real WSN where National Instruments (NI) WSN kit [62] is used to examine the effectiveness of our approaches.

The possible applications of WSN in the field of agriculture especially for greenhouses are investigated. In addition, a prototype for a WSN plus fuzzy logic controller to monitor and detect the diseases based on the environmental parameters such as temperature, humidity, soil moisture, etc is proposed.

1.2. Thesis's Organization

This chapter gave a brief introduction on wireless sensor networks. In addition, some of the QoS parameters are defined. Moreover, the objectives of this thesis are indicated as maximizing the network lifetime as well as proposing a novel technique for detecting the plants diseases inside greenhouses. Chapter 2 introduces the WSN in more details indicating the technologies used in WSN operations and explaining the architecture of the communication protocols and other functions such as localization, security, and power management. Chapter 3 deals with the power conservation techniques while chapter 4 explains our proposed solution based on the *Exponential Smoothing Predictors* and *Fuzzy Logic Algorithm*. Chapter 5 introduces the simulation and experimental results. Afterwards, chapter 6 describes the proposed applications in the field of agriculture. Moreover, our proposed system for detecting the plants diseases based on WSNs, REDR technique, and the Fuzzy Logic algorithm is introduced. Finally, chapter 7 draws conclusions for this work and presents a comparison between the different approaches proposed in this thesis and describes the future work that can enhance the performance of our work.

Chapter 2

Wireless Sensor Networks Overview

2.1. Introduction

The study of wireless sensor networks is a challenge in that it requires a deep knowledge from different disciplines. In this chapter, some of these disciplines including the wireless technologies, the communication protocols, localization algorithms, coverage, storage, synchronization, security, and data aggregation and compression will be briefly discussed.

WSN is an infrastructure-less network consists of smart nodes with multiple on-board sensors networked through wireless lines as well as the Internet and deployed in large numbers. These networks have three missions: sensors generating data, sinks gathering the data, and users that send queries and receive data via the sinks. Each node is able to be autonomous because it had an onboard microcontroller, memory, radio transceiver, and sensors as shown in Fig. 2.1.

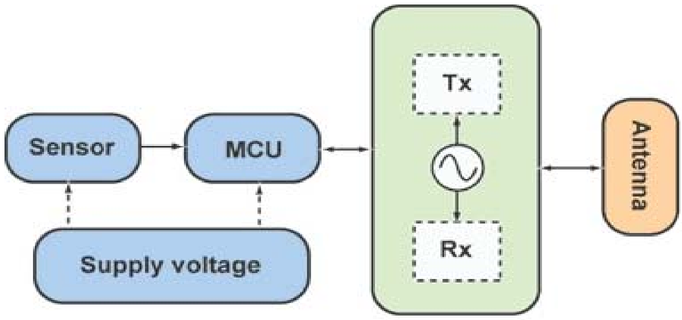


Fig. 2.1 The main subsystems in the sensor nodes

WSNs have many advantages over the other monitoring systems such as the low cost deployment, the ease of scalability, the higher robustness against failures through distributed operation, re-configurability, self-organization, limited size, minimal memory, and good computation ability, as well as their cost. On the contrary, sensor nodes are battery powered however there is research being done on using solar power as a power source [8]. This limited supply of power makes power consumption a critical issue in WSNs especially when the network is deployed in difficult-to-access areas. WSNs have the following unique characteristics [87]:

1. **Real time requirements**: in many applications, it is mandatory to produce an instant action. For instance, fire detection is an application that requires real time sensing and acting to prevent the fire from being uncontrollable. Also, in health monitoring applications it is required to sense and report the problem in very short time to help doctors deal with this problem as fast as possible.

2. **Homogeneous network:** the network is composed of sensor nodes that are identical where they have the same processing capabilities and memory size. However, small number of nodes with enhanced capabilities may be used to increase the monitoring ability.

3. **Relatively dispersed network:** like ad hoc networks, the sensor nodes are randomly deployed in relatively large geographical areas. Therefore, one hop communication among the sensor nodes is difficult.

2.2. WSN Classifications

WSN can be classified as *unstructured* and *structured networks* [42]. In the former type, a huge number of sensor nodes are randomly deployed into the monitored field. Therefore, performing maintenance such as detecting failures and managing connectivity is difficult due to the dense deployment.

In the structured networks, sensor nodes are deployed with predefined plan that makes the maintenance easier due to deploying smaller number of nodes. The architecture of WSNs can be classified as *Automated Architecture* and *Semi-Automated Architecture* as shown in Fig. 2.2. In the first one, sensor nodes forward their measurements to actuators in order to make the suitable actions. In the second type, sensor nodes send their readings to a central unit, which in turn collects these data and then commands the actuators to execute the suitable action.

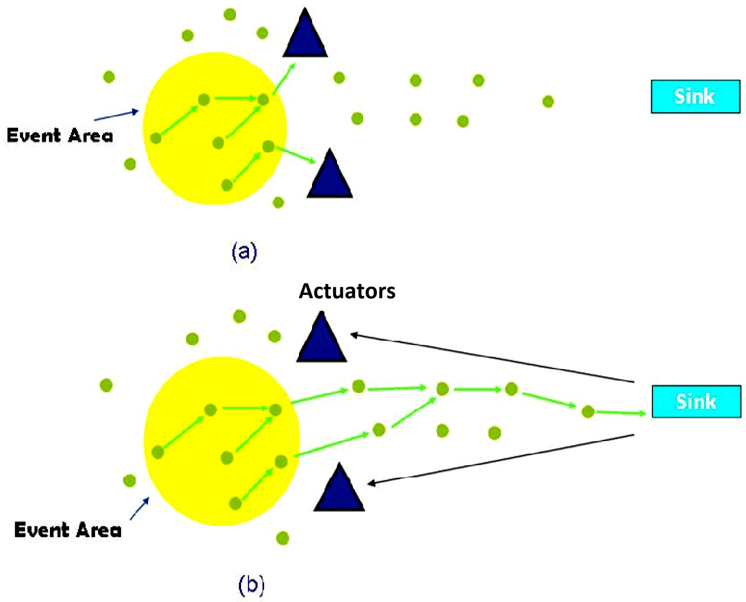


Fig. 2.2 (a) Automated and (b) Semi-Automated Architecture [101]

The WSN standards are defined for the *semi-automated architecture* [50]. However, the *automated architecture* has low latency, longer network lifetime since it avoids flooding the network with messages transmitted from sensor nodes to the sink node or from the sink node to actuators. A different view shows that there are many applications require aggregating the measurement inside a central unit such as the military applications and health monitoring. In these applications, the human intervention is necessary for producing the suitable action.

In addition, WSN can be classified according to the surrounding environment to *Terrestrial*, *Underground*, *Underwater*, *Multimedia*, and *Mobile WSN* [50]. In the *terrestrial* type, the network is composed of thousands to hundreds of thousands of sensor nodes deployed on land where the nodes can be equipped with additional energy source such as solar cells. While in *underground* WSN, the network is deployed in caves, mines, or underground whereas sink nodes are deployed above ground to report the measurements to the user. In the next type, sensors are deployed into oceans environment. On the other hand, *multimedia* networks use devices that are able to store, process, and retrieve multimedia data including audio, video, and images. The last type uses devices that have the ability to move and still has reliable communication with other devices.

The main challenges related to WSN implementation are the following [71]:

1. **Energy conservation:** it is required to minimize the size of sensor nodes, which in turn reduces the capacity of the battery. Therefore, the available energy is limited. However, it is necessary to develop methods that maximize the energy efficiency.

2. **Low-quality communications:** the quality of radio communication is extremely influenced by the surrounding environment. However, WSN is often used in harsh environments such as oceans, forests, rivers, and volcanoes. As result, the quality of radio communication among the nodes is decreased. For instance, underwater WSNs use acoustic waves for wireless communication but these waves suffer from the limited

bandwidth, long propagation delay, and single fading. Consequently, the performance of the network is reduced.

3. **Resource-constrained computation:** it is required to maintain the QoS requirements in spite of the few available resources. For instance, multimedia WSNs require high bandwidth, high-energy consumption, QoS provisioning, and high data processing and additional compressing techniques. Therefore, achieving these requirements represents a challenge due to the limited resources allocated for each node.

4. **Data processing:** despite the scarce of the available resources, it is necessary to compress and aggregate the data collected from other sensor nodes according to the multi-hop communication technique. Moreover, sensor nodes may provide different level of compression according to the needs of the user.

5. **Lack of easy-to-commercialize applications:** most WSN application scenarios are very specific, and a company would have little or no profit in developing an application for a very specific scenario since the potential buyers would be very few.

2.3. WSN applications

Applications of WSNs can be categorized into two main scenarios [67]. In the first one, the function of the network is to detect a certain event such as fire and earthquake whereas the function of the network in the second type is to estimate a random spatial or temporal process.

In general, the computational capabilities, the radio transceiver, and the sensing units included in WSN open the door to thousands of applications in all fields around us such as environmental monitoring, food-safety, habitant monitoring, inventory control, health care, mood-based services, positioning and animals tracking, energy management, entertainment, logistics, transportation, homeland security, and military initiatives.

For instance, environmental monitoring is one of the most important applications of WSN in which the deployed sensors obtain localized measurements and detailed information about natural spaces where it is not possible to do this through known methods. Often in these applications, WSN provides solutions for security and surveillance concerns. For example, underwater sensor networks are used for long term monitoring of coral reefs as proposed in [41]. Moreover, natural disasters such as flooding and earthquakes may be early predicted through deploying WSN that responds to the changes of the environment quickly with high degree of security. In these cases, the WSN lifetime must be as long as possible especially in unreachable areas.

In recent years, Portugal has had serious problem with forest fires since it becomes very difficult to resist the fire when it becomes large [78]. Therefore, they used WSN to collect information such as temperature, humidity, pressure, and position. Afterwards, the data are sent to a control centre via a number of gateway devices. As soon as fire-related event is detected, the firefighters and helicopters can be sent to put out the fire before it grows. More information about the use of WSNs in detecting forest fires can be found in [19], [54], and [56]. Similarly, WSN can be useful in detecting and localizing damages in buildings, bridges, ships, and aircrafts through measuring the structural responses to ambient or forced excitations such as earthquakes, winds, and vehicles.

In the medical field, WSN are used for remote monitoring of human physiological data, tracking and monitoring of doctors and patients inside a hospital, drug administrator in hospitals, etc [47]. In this kind of applications, the monitoring system must be safe and reliable. In other words, it should require minimal maintenance, energy-harnessing body heat, and the transmission between the nodes must be reliable.

In the field of transportation, WSN can add more comfort and safety to transportation through developing an autonomous transportation system, which can be implemented with different scenarios. For instance, in one scenario cars may communicate with each other in order to organize traffic, whereas in other one traffic conditions of roads may be monitored by installing static entities along highways.

In the field of agriculture, WSN can be used in different situations. For instance, Bermuda grass is a plant that is used in Stadiums, clubs, and tourist villages. This plant is sensitive to temperature degrees where it is subject to yellowing when the surrounding environment has small temperature degree. Consequently, WSN can be used to indicate the changes in temperature in order to protect this grass from yellowing. For example, stadiums located near the seashore are subject to changes in temperature degrees all day. Here, WSN may be used to help workers in making certain decision about the grass in order to protect it from the great changes in the temperature degrees.

The types of plants can be divided into two categories. One of them is the plants that grow in the winter while the second type grows in the summer. Therefore, there is a need for storage system that enables us to have these products all the year. However, some of the plants need small temperature during storage phase such as apple where its optimum storage temperature degree is ranged from 0 to 3°C. Others require relatively high temperature degrees during storage phase such as banana and Guava where their optimum storage temperature degree is ranged from 13 to 15°C [64]. WSN can be used to ensure the actual temperature inside the storage areas to allow the user to make a decision if the temperature degrees decreased or increased beyond the mentioned ranges. In addition, the storage place sometimes has a large area, which allows the contrast of the temperature degrees inside this place. Therefore, many sensor nodes can be used inside this area to detect the points at which there is a probability of danger.

Another application is the drying that is the most common method of medicinal plant preservation. The drying regime has significant impact on the drug quality and consequently earnings. Conventionally, low drying temperatures between 30 and 50°C are recommended to protect sensitive active ingredients [79]. So, WSN can be used to keep temperature within this range.

The comfort zone (CZ) is the range of temperature degrees and humidity percentages within which people feel comfortable under calm wind conditions. Generally, as long as temperature increases, tolerance to humidity decreases, and vice versa. For people adapted to most temperate regions, CZ has air temperatures of 20-25°C and relative humidity percentages between 25 and 75 per cent. In Britain, Scientists found that the optimum conditions for comfort are 15°C and 60 per cent relative humidity. The same concept applies to animals and fish. So to maintain the environment of the farms within CZ of animals or of fish, WSN can be used to maintain these parameters within the specified range. In addition, WSN can be used for other monitoring activities including detecting the heat period (missing the day where a sow can become pregnant has a major impact on pig production) and possibly detecting illness (such as a broken leg). Similar approach is proposed in [72] in which mobile WSNs are used to track animals' migrations.

2.4. Protocol Stack For WSN

The protocol stack for WSN is much similar to the traditional computer networks protocol stack. As shown in Fig. 2.3, the WSN protocol stack consists of the *physical layer*, *Data link layer, network layer*, *transport layer*, *application layer*, *power management plan*, *mobility management plan*, and *task management plan* [37].

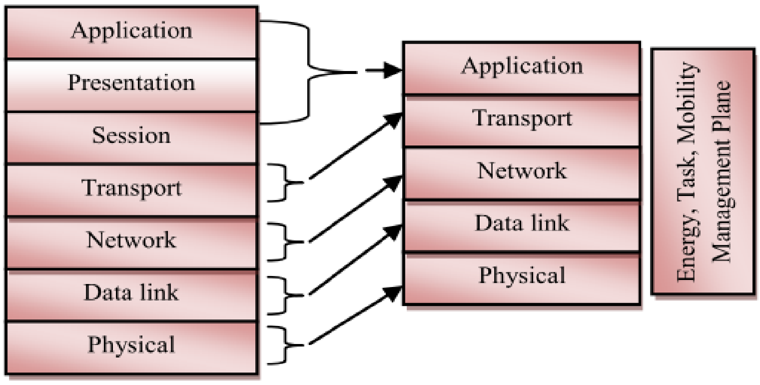


Fig. 2.3 Protocol stack for WSNs [53]

The physical layer is responsible for frequency selection, carrier frequency generation, signal detection, modulation, encryption, transmission, and receiving radio messages. Moreover, it is concerned by minimizing energy consumption. For instance, the minimum output power required to transmit over a distance d is increased from d^2 to d^4 when the antenna are near the ground as is typical in WSNs. This is due in part to ground-reflected rays, which causes partial signal cancellation. Fortunately, this problem is overcome by multi-hop communication and high node density.

The data link layer is responsible for multiplexing of data streams, data frame detection, medium access, and error control. Dedicated Medium

Access Control (MAC) protocol should be built in order to handle the issues of power conservation and data-centric routing. Additionally, the MAC protocol has to establish communication links between possibly thousands of nodes, provide the network self-organizing capabilities, and efficiently share communication resources between all nodes. However, it should take into account the permanent changes in the network topology due to node failure and redistribution. Examples of these protocols are Sensor-MAC (SMAC), Etiquette Protocol, and Carrier Sense Multiple Access (CSMA).

The Link layer answers how two nodes can talk to each other? On the other hand, the network layer is responsible for deciding the next nodes to talk with. Flooding is the simplest method in which each node receiving data repeats it by broadcasting the data to every neighbor unless the max hop lifetime of the data has been reached or the receiving node is the destination. Unfortunately, it suffers from many drawbacks such as implosion that occurs when two nodes (A and B) share multiple (n) neighbors. Node A will broadcast data to all of these neighbors. Node B will then receive a copy of the data from each of them. In addition, overlap may occur when two nodes share the same sensing region. If a stimulus occurs within this overlap, both nodes will report it. Therefore, gossiping is used in which nodes are randomly chooses a neighbor and sends the data to it. This method avoids implosion, but still suffers from overlap. Therefore, data aggregation techniques are proposed to solve the implosion and overlap problems. In these techniques, the sink asks sensor nodes to report the ambient condition of the phenomena. Data received from multiple sensor nodes are aggregated as if they are about the same attribute of the phenomenon when they reach the same routing node on the way back to the sink. More detail about these methods can be found in [97], and [37].

The transport layer is responsible for the communication with the outside world [53]. Sending data from the sink node to outside user is a problem because WSNs use attribute based naming instead of global identification for transmitting the data. In addition, transport layer ensures the reliability and the quality of data at the source and the sink. To perform these missions, the protocols should support multiple applications and efficiently handle variable reliability. Moreover, they have to be able to detect packet loss and contain congestion control mechanism.

In the application layer, many protocols are proposed to make the hardware and software of lower layers transparent to the sensor network management applications such as Sensor Management Protocol (SMP), Task Assignment and Data Advertisement Protocol (TADAP), and Sensor Query and Data Dissemination Protocol (SQDDP) [42]. For instance, SMP enables the interaction between applications and sensor networks through setting rules for handling [21]:

- Data aggregation, attribute-based naming, and clustering,
- Exchange data related to the location finding algorithms,
- Time synchronization,
- Moving sensor nodes,
- Turning nodes on or off,
- Querying WSN configuration status, reconfiguring the WSN,
- Authentication, key distribution, and security.

The function of the power management plane is to minimize the power consumption by different means to increase the network lifetime. On the other hand, the mobility management plane handles the operations needed for the movement of sensor nodes and it maintains a data route to the sink. In addition, it maintains a route to user and it Keeps track of neighborhood topology. Finally, the task management plane carries out the sensing task assigned to specific sensor nodes while other nodes focus their energy resource on routing and data aggregation. Detailed information about the stack of WSN and the different protocols used can be found in [50], [34], and [42].

2.5. Wireless Technologies for WSNs

Some hardware platforms are specialized for optimizing only one feature (e.g., high data rate, long transfer range, or low-power mode). However, the most restrictive parameters for WSNs are both power consumption and distance. In the following paragraphs, different technologies that can be used for WSNs will be briefly discussed.

The *Bluetooth* wireless communications technology provides a personal area network (PAN) for exchanging data between Bluetooth-capable devices within certain proximity [9]. *Bluetooth* technology has a low-power mode and high-integrated devices that uses the 868 and 915 MHz and the 2.4 GHz radio bands to communicate at 1 Mb per second., but it is limited to short-distance communications [76]. Therefore, it can be useful only when the network is densely deployed and the nodes connected together through multi-hop communications. The typical features of Bluetooth are [103]:

- **Ubiquity:** when sensor nodes are equipped with Bluetooth, they can communicate with gateways supported by Bluetooth interfaces that are widely spread. Additionally, WSN uses actuators to perform actions based on the measured phenomena. Therefore, commercial products with Bluetooth interface can be used as actuators. For instance, when Bluetooth is used as the radio for sensor nodes, easy communication with personal devices such as PDAs and laptops can be achieved.

- **Power efficiency:** Bluetooth allows sensor nodes to enter low power modes while maintaining synchronization during times of no transmission or reception on the communication link. These modes greatly reduce the power consumption.

- **Interference Resilience:** since Bluetooth exploits frequency-hopping technique. Therefore, sensor nodes within communication range can use separate channels to transmit data.

Another technology that can be employed is the *Wavenis* technology that provides long-range data connections and services for autonomous devices with extremely limited battery resources [3]. Moreover, it is intended for Ultra Low Power Systems (ULPS) and long-range wireless communications. Wavenis extends the industry standard Bluetooth protocol to provide robust wireless solutions for building ad hoc and fixed networks using autonomous battery-powered devices. Its architecture consists of an RF transceiver and a protocol stack, both specifically adapted to provide the optimal combination of secure and reliable connections, long range, minimal power consumption, and highly resistant to interference and obstacles. Wavenis uses three frequency bands 868, 915, and 433 MHz, with data rates between 2.4 and 100 kbps (typical 10 kbps), with a high line-of-sight range of up to 1 Km [99]. In addition, it uses different techniques for saving power including data interleaving, frequency-hopping spread spectrum (FHSS), forward error correction, and Gaussian Frequency Shift Keying modulation (GFSK).

The third technology is the *Ultra Wideband* (UWB) in which information is transmitted at a large bandwidth [45]. As a result, the width of the pulses will be very small typically 1 to 2 nanoseconds. The UWB signals occupy the frequency band from 3.1 to 10.6 GHz. Moreover, they have immunity against interferences due to the use of spread-spectrum modulation techniques. More details about UWB can be found in [45] and [88].

The fourth one is the *Wibree* technology that is a short-range wireless communications protocol [63]. This technology can operate with either a standalone chip or a dual-mode chip. The standalone chip is a small device able to operate with very low power consumption, dual-mode Bluetooth Wibree is able to communicate with Bluetooth standard devices with less power consumption and at distances of 5 to 10 meters using the 2.5-GHz band. The main characteristics of Wibree are the ultra low-power idle mode operation, power-saving technology, device discovery, reliable point-to-multipoint data transfer, and encrypted communications. There are similarities between Wibree and Bluetooth standard such as the data rates where they use the 2.45 GHz band to transfer data and have a 1 Mbps transfer rate and a rage of about 10 m. On the other hand, they differ in size, price, and most of all power consumption where Wibree is designed to consume only a fraction of the power Bluetooth chips existed nowadays. Moreover, it extends the application that can be made by the Bluetooth including audio and video monitoring to other applications that require the transmission of small amount of data [100].

The final one is ZigBee which is communication standard based on the IEEE 802.15.4 [43] standard for wireless personal area networks (WPANs). It provides a long battery life and has a lower-cost alternative to Bluetooth. The IEEE 802.15.4 standard defines two layers, the MAC, and the physical layer (PHY). The physical layer uses three frequency bands to transfer data 2,450 MHz, 915 MHz, and 868 MHz with data rates 250 kbps, 40 kbps, and

20 kbps respectively [107]. Moreover, this layer is responsible for channel selection, link quality estimation, energy measurement and clear channel assessment. ZigBee standardizes both the application and the network layers. Whereas, the network layer is responsible for organizing and providing routing over a multi-hop network, specifying different network topologies. On the other hand, the application layer provides the necessary protocols for the distributed application development and communication. There are three types of ZigBee devices: ZigBee coordinator, ZigBee router, and ZigBee end device. ZigBee coordinator handles network formation, stores information, and can bridge networks together [106] while ZigBee routers link groups of devices together and provide multi-hop communication across devices. Finally, ZigBee end device consists of sensors, actuators, and controllers that collects data and communicates only with the router or the coordinator.

2.6. Localization in WSN

The sensor nodes' measurements may be valueless if the locations of the sensor nodes are not clearly defined. However, Location often not known a priori, therefore, localization is the task of determining the position of a sensor or the spatial relationships among objects. Localization can be classified into *Range* method, and *Range-Free methods*.

The idea behind the range method is to estimate the distances or angle between node pairs, and then compute the position of individual nodes in the global network. Triangulation is for example the most basic approach for computing the position.

Many techniques are proposed for Ranging such as Time of Arrival (TOA), Time Difference of Arrival (TDOA), Angle of Arrival (AOA), and

Received Signal Strength (RSS). In TOA method, the distance between transmitter and receiver is determined using the measured signal propagation time and known signal velocity (radio signals: 300kmps, i.e., approx. 30ns to travel 10m).

$$d = (t_2 - t_1) * v \tag{2.1}$$

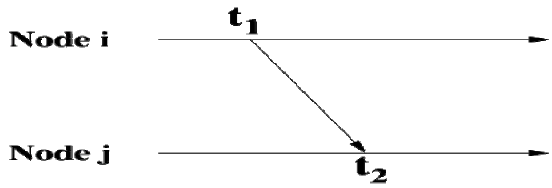


Fig. 2.4 Time elapsed during signal propagation [55]

The advantage of this method is simplicity since it does not require additional hardware installation. On the other hand, it requires an accurate synchronization between the sender and the receiver clocks. Synchronization adds cost and complexity to the WSN [55].

For instance, GPS-Based Localization provides framework for determining geographic positions. The levels of services that can be obtained are classified into Standard Positioning Service (SPS) and Precise Positioning Service (PPS). In the first type, the service is provided to all users with no restrictions or direct charge. In addition, SPS provides high accuracy when sensitive receivers are used. Better accuracy is achieved by the second type that is used only by the U.S military, and U.S. Federal Government users. It relies on transmitting two signals to reduce transmission errors.

In the TDOA method, two signals with different velocities are used [105]. For instance, radio signal is sent at t_1 and is received at t_2. Afterwards, acoustic signal is transmitted at t_3=t_1+t_{wait} and is received at t_4. In this case, no clock synchronization is required and the distance measurements can be very accurate but it requires additional hardware.

$$d = (v_1 - v_2) * (t_4 - t_2 - t_{wait}) \tag{2.2}$$

In the AOA method, two techniques can be considered. The first one makes use of the receiver antenna's amplitude while the second one makes use of the receiver antenna's phase response [55]. The accuracy of AOA measurements is influenced by the directivity of the antenna, the shadowing effect, and multipath reflections.

In the RSS method, the distance is determined based on the attenuation of the transmitted signals. Therefore, By knowing the original emitted power and comparing it to the received signal power, one can estimate the attenuation g and deduce the distance via, for example, a free space path-loss model:

$$g = d^{-\alpha} \tag{2.3}$$

With the exponent α changes from 2 to 4 according to the surrounding environment. The problem with this method is the multipath fading. Therefore, any reverberations of the signal will influence the received signal strength. Consequently, it has to be measured at the appropriate moment. Some proposals consider the first peak, whereas others prefer an average of the first periods.

For the Range-Free method, the algorithm is based on the connectivity information that can be measured by counting the amount of received packets. Therefore, when two nodes are communicating then the distance between them is less than their maximum transmission range with great probability [28]. The main advantages of range-free method are its simplicity and low cost. Therefore, they are suitable to applications where location accuracy is less critical. However, the localization error depends on the density of nodes, on the number of anchor nodes and on the network topology. More detail can be found in [28], and [67].

2.7. Time Synchronization

No global clock or common memory exists in the distributed systems where each node has its own internal clock. These clocks drift through the day even if they were synchronized when they start. As a result, serious problems appear to applications that depend on a synchronized notion of time. Therefore, high-precision synchronization mechanisms must be provided to compensate for these inaccuracies.

For most applications and algorithms that run in a distributed system, it is necessary to know the time of the day at which an event occurred on a specific node and the time interval between two events that occurred on different nodes in the network [7].

Many synchronization protocols are developed for the wired network. However, these protocols are unsuitable for WSNs due to the dense deployment of sensor nodes, self-configuration, and energy constraints. Therefore, new protocols are developed to cope up with the WSNs constraints such as Remote Clock Reading method (RCR), Time Transmission Protocol (TTP), Offset Delay Estimation method (ODE), and Set-Valued Estimation method (SVE). These protocols are designed to satisfy the following requirements:

- It has to cope up with unreliable network transmission.

- The protocol should provide each node with the local time on the other nodes' clocks.
- The protocol should ensure that the clocks are advanced rather than being outright set back until the correction is achieved.
- The synchronization overhead must not influence the system performance.

The correction of the clocks drift can be classified according to the nature of application to either precise synchronization or relative synchronization. In the former type, clocks are synchronized to an accurate real-time standard like Universal Coordinated Time (UTC). While, the latter type is used in applications that require only the logical condition that no two processes access the critical section concurrently. Therefore, clocks are relatively synchronized to each other since the requirement is only to provide an ordering of events, and not the exact real-world time at which each event occurred. Detailed information about the synchronization protocols can be found in [7], [70], and [48].

Chapter 3

Problem Statement and Related Work

3.1. Problem Statement

Since each node has limited battery power, efficient power consumption is a challenging problem in WSN. During signal propagation, the signal decays as $r^{-\alpha}$ with transmission range r, where α is the loss exponent of the signal [90]. The limited power and signal loss during propagation add fundamental constraints on the operational lifetime of WSN.

In many applications, it is expected that each sensor node last for a long time because in most of the cases WSNs are usually used in remote areas; recharging and replacing power supply units is almost impossible especially with the large number of nodes. For this reason, energy efficient models have to be employed to reduce the wasteful power that is consumed in the radio communication. Therefore, an optimal solution that maximizes the lifetime of the whole network while reducing unnecessary power consumption has to be found.

In order to maximize the network' lifetime, the power dissipation characteristics of a wireless sensor node need to be analyzed. Therefore, awareness of the main energy sources in WSN as well as the subsystems that consume this energy is required. As shown in Fig. 3.1, the sensor node consists of four main units:

- **The processing unit**: Microcontroller unit provides the sensor node with intelligence where it is responsible for controlling the node, and executing the communication protocols and the signal processing algorithms on the gathered sensor data [93]. Other alternatives that can be employed as a controller are general-purpose microprocessor, Digital Signal Processors (DSP), Field Programmable Gate Array (FPGA), and Application-Specific Integrated Circuit (ASIC). Each one has its own advantages and disadvantages. However, microcontroller is the suitable device for sensor nodes since it is flexible to connect to other device, programmable, consumes small power, and supports various operating modes, including Active, Idle, and Sleep modes. The choice of the type of the microcontroller depends on the application scenario. For instance, the ARM microcontroller from Intel consumes around 400 mW of power while executing instructions, whereas the ATmega103L AVR microcontroller from Atmel consumes only around 16.5 mW, but provides much lower performance [93]. Therefore, depending on the nature of the application, the microcontroller is selected to satisfy the required performance.

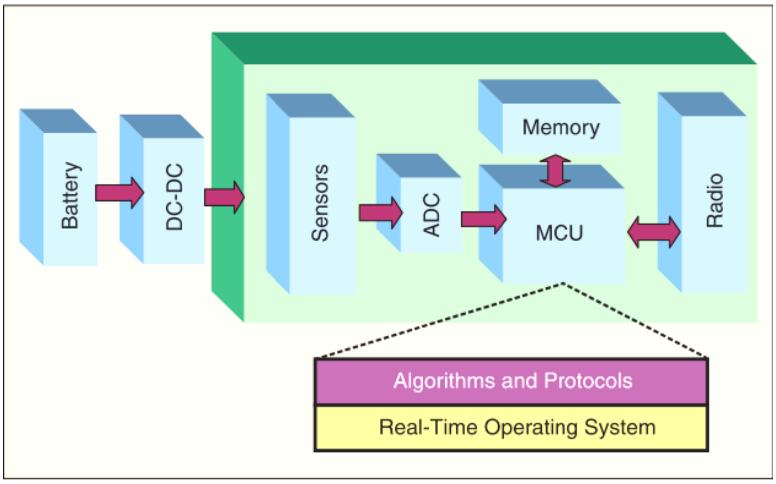


Fig. 3.1 The architecture of a sensor node [93]

- **The sensing unit:** sensors are used to detect the desired phenomena from the surrounding environment such as Capacitive Sensors, Inductive Sensors, Magnetic and Electromagnetic Sensors, Thermal Sensors, Thermocouples, Optical Transducers, Chemical and Biological Transducers, and Acoustic Sensors [24]. The output signal from sensors are digitalized by Analog-to-Digital Converter (ADC) and then sent to the microcontroller for more processing. The power consumed in these sensors is spent in signal sampling, conversion of physical signals to electrical ones, signal conditioning, and analog-to-digital conversion. Sensors are classified into Passive-Omni Directional Sensors, Passive-narrow-beam sensors, and Active Sensors. In the first type, sensors are self-powered and do not manipulate the surrounding environment by active probing. Furthermore, they sense the phenomena in all directions. In addition, the second type is the same as the first one except that the measurement is achieved in a certain direction. Finally, the third type interacts with the surrounding environment such as Radar sensors. Therefore, active sensors are more consumer of power than passive sensors. Recently, Micro electro mechanical Systems (MEMS) sensors are very well developed and are available for most sensing applications in wireless networks [77].

- **The radio communication unit:** The radio transceiver connects each node with its colleagues in the network where the WSN' nodes use the communication frequencies between about 433 MHz and 2.4 GHz. The power consumption of a radio transceiver is influenced by the type of modulation scheme used, the data rate, the transmit power, and the operational duty cycle. The transceiver has four modes of operations, which are Transmit, Receive, Idle, and Sleep. The transition among these modes costs a significant amount of energy. However when the transceiver is not transmitting or receiving, it is more power saving to switch to the Sleep mode rather than going into the Idle mode. Other alternatives for the wireless transmission media are the Optical communication (Laser) and, Infrared. However, both of them have a problem that stands against using it in the platform of WSNs. For instance, Laser needs line-of-sight for communication and sensitive to atmospheric conditions while Infrared is limited in its broadcasting capacity.

- **The energy source:** this source of energy is used to feed up other subsystems. Batteries are the usual energy source used in WSN. Their operation depends on many factors such as battery dimensions, type of electrode material used, and diffusion rate of the active materials in the electrolyte [93]. Batteries are classified into chargeable and non-chargeable. In addition, they can be classified according to the electrochemical material used for electrode including Nickel-Cadmium (NiCad), Nickel-Zinc (NiZn), Nickel Metal Hydride (Nimh), and Lithium-Ion.

One of the limitations of WSNs is the requirement of designing nodes with small size. This provision limits the allowed size for the battery. As a result, the node' lifetime is reduced. For instance, the non-rechargeable lithium batteries can provide 2880 J/cm^3. Suppose that an electronic device with a 1 cm^3 will consume 100 μW of power on average. Then, this device will survive for approximately one-year [86]. However, this lifetime is not sufficient compared to the required performance.

Power available at each sensor node is consumed in the sensing, the processing, and the radio communication operations. However, research studies showed that the wireless communication is the dominant factor of power consumption in sensor nodes. They indicated that the energy cost of transmitting a single bit of information is approximately the same as that required for processing a thousand operations in a typical sensor node [29]. On the other hand, it is found that the power consumed in the sensing operations vary according to the application. Overall, there is an urgent need for developing alternative sources of power in WSNs as well as developing techniques for power conservation.

In this thesis, different methods of energy saving and energy harvesting are investigated. Additionally, novel method is proposed to minimize the energy consumption by reducing the amount of data transmitted among the nodes as well as from the sensor nodes to the sink node. Two algorithms are utilized to achieve this method, which are exponential smoothing predictors and fuzzy logic controllers.

3.2. Related Work

Many studies tried to find an optimal solution for the powering problem. For instance, one proposed solution is to improve the density of the energy storage subsystem in order to provide larger amount of energy for the sensor nodes [23]. These method promise to extend the lifetime of sensor nodes, they cannot extend their lifetime indefinitely.

Another proposed solution is to distribute power to sensor nodes through RF radiation [86]. Unfortunately, this method is not practical due to the dense deployment of sensor nodes. In addition, the amount of radiation needed to empower the nodes in small space like a room would probably present a health risk and today exceeds the Federal Communications Commission (FCC) regulations.

The problem of WSN lifetime maximization, in general, has been addressed in several other works. Hnin Yu et al [37], for instance, listed four approaches for saving energy. The first one is the use of sensors' scheduling by which sensors alternate between sleeping and waking; the waking sensors sense events in their environments and the sleeping sensors avoid idle listening and overhearing. The problem with such approach is that it requires synchronization among sensors, which generates overhead messages. In addition, it might not be possible to do synchronization especially in mobile WSN. The second lifetime maximization technique is the in-network processing where intermediate nodes may aggregate several events into a single event to reduce transmissions. Again, this technique is perfect only when sensors' readings do not vary and readings accuracy is not that important.

Network coding is the third lifetime maximization technique in which the collected data are mixed at intermediate node then encoding packets are sent instead of sending individual packets; consequently reducing the traffic. In the fourth approach, data collision is avoided to reduce the retransmission of packets; this is achieved by employing communication protocols including Time Division Multiple Access (TDMA), Frequency Division Multiple Access (FDMA), and Code Division Multiple Access (CDMA). The idea behind these protocols is to avoid interference by scheduling nodes onto different sub-channels that are divided by time, frequency, or orthogonal codes.

K. Kalpakis et al [52] proposed a technique for maximizing the WSN lifetime using the polynomial–time algorithms to find an efficient manner in which the data should be collected from all sensors and transmitted to the base station. S. Lindsey et al [83] proposed another technique for saving the energy in which a linear chain of all the nodes are formed to gather data, and nodes took turns to transmit to the base station. Other approaches were proposed such as the dynamic voltage scaling, dynamic frequency scaling, energy efficient routing, asynchronous processors, nodes partitioning (clustering), the use of ultra wideband for radio communication and the use of CMOS low voltage and low power wireless IC. These approaches focused on the networking management and the sensing operations to solve the problem of power consumption in WSNs so that the network' lifetime is maximized.

In the following sections, a number of the power saving techniques will be discussed in more details. As can be seen in Fig. 3.2, the power management approaches are classified into energy scavenging and energy conservation methods. The latter type is also categorized into duty cycling methods and data driven method.

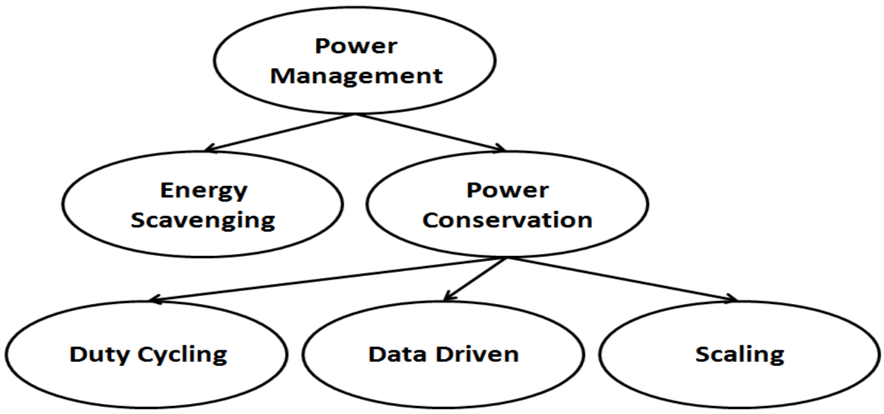


Fig. 3.2 Taxonomy of power management approaches

3.2.1. Energy Scavenging

The idea behind this method is to convert the ambient sources of energy into electricity to be used for empowering sensor nodes [12]. Therefore, these methods grasp many researches to study the approaches that can be used to deliver power from the surrounding environment using energy scavenging or harvesting techniques to the electronic components inside the sensor nodes. The harvesting techniques rely on extracting energy from the surrounding vibration/motion, thermal gradients, solar energy, and acoustic waves and then deliver energy directly to the wireless sensor load or to a storage element such as a rechargeable battery or capacitor.

A summary review of the major energy scavenging techniques including the solar energy, vibration, and temperature variation is presented below:

3.2.1.1. Solar Energy

Solar energy is the most common method for harvesting energy in WSNs. Although, it provides relatively high efficiency only when there is sufficient sunlight or artificial light. Moreover, it exhibits a strong nonlinear electrical characteristic. The illumination level and the solar cell technology are parameters that highly affect the power available from solar cells [49]. For polycrystalline silicon and gallium indium phosphide, the efficiency is 15% during outdoor illumination level of 5000 W/m^2. However, their efficiency is reduced to 10% during indoor illumination levels of 10. Similarly, the efficiency of the amorphous silicon is ranged from 2–5% during outdoor condition while this efficiency is minimized to reach 2% during indoor condition [10]. Studies are made to ensure that energy is not lost during the transfer from the harvester to the electronic devices where they proposed a low-power maximum power point tracker (MPPT) circuit to transfer the generated energy to batteries or capacitors efficiently [10].

3.2.1.2. Energy from Vibration and Movement

Vibrations are found in many environments including automotive, buildings, bridges, railways, industrial machines, household appliances, etc.

Through using a suitable mechanical-to-electrical energy converter, the vibration can be converted to electricity that is able to empower the electronic components inside the sensor nodes [10]. These converters rely on electromagnetic, electrostatic, or piezoelectric principle in the conversion process.

For the electromagnetic generators, the energy in the surrounding vibration is employed to generate a magnetic mass that cuts the turns of a coil, thus inducing a voltage and causing a current to flow in the attached electrical load. On the other hand, the piezoelectric properties of some materials may be exploited to generate a voltage when the piezoelectric generator is exposed to stress through the surrounding vibration. For instance, traffic sensors can be partially empowered by the short duration vibrations when a vehicle passes over the sensor [51]. Additionally, it is found that 12-bit digital word information can be send wirelessly using the amount of energy that is generated when a piezoelectric pushbutton is depressed [98]. On the other hand, the electrostatic generators make use of the vibration force in order to push and pull the plates of a charged capacitor against the force of electrostatic attraction. This movement forms a change in the electric field consequently producing energy that is stored in the capacitor.

3.2.1.3. Energy from Temperature Variations

In this approach, the energy is provided by means of converting the ambient temperature variations into electricity. For example, devices such as Atmos clock can be used to perform the conversion where it includes a sealed liquid [86]. This liquid is vaporized at temperature of 21°C. Consequently, the resultant gas increases the pressure on a spring that winds the clock. However, the level of power output is still substantially lower than other possible methods. The scavenging techniques are summarized in Table 3.1. Pure power scavenging sources are given in this table and thus the amount of power available is not a function of the lifetime of the device.

Table 3.1 Comparison of energy scavenging techniques [86]

Energy Source	Power Density (μW/cm 3) 1 Year lifetime
Solar (Outdoors)	15,000 - direct sun, 150 - cloudy day
Solar (Indoors)	6
Vibration	200
Acoustic Noise	0.003 @ 75 dB 0.96 @ 100 dB
Daily Temp. Variation	10
Temperature Gradient	15 @ 10 ºC gradient
Shoe Inserts	330

3.2.2. Energy Conservation Methods

In this section, approaches for minimizing the power consumption consequently maximize the network' lifetime are addressed. Scaling approaches, duty cycle methods, as well as data driven methods will be discussed before focusing in the proposed solution.

3.2.2.1. Scaling Methods

In this approach, a certain parameter is made dynamic in order to minimize the consumed power. As shown in Fig. 3.3, scaling methods are categorized into three types: Dynamic Voltage Scaling (DVS), Dynamic Thermal Management (DTM), and Dynamic Modulation Scaling (DMS). In the first type, the CPU energy is reduced by dynamically adjusting the clock speed and supply voltage based on the instantaneous workload thus when the workload is low, the supply voltage is scaled down to reduce the energy consumed [40]. The same principle applies for the frequency where energy can be saved by reducing the operation frequency during the period of reduced activity. However, slight degradation for the system performance occurs. Therefore, the tradeoff has to be considered between the reduction in power consumption and satisfying higher performance. This trade off can be achieved by tuning supply voltage to deliver just the required performance.

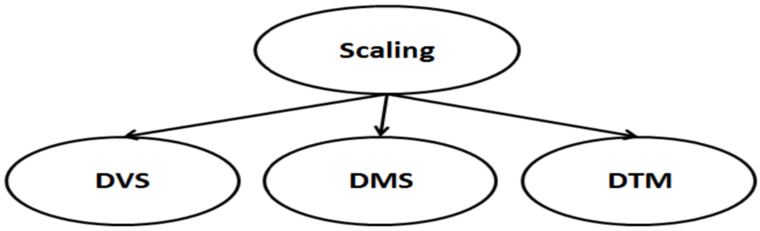


Fig. 3.3 Taxonomy of scaling techniques

For example, this technique is examined by using SA-1100 DVS-enabled microprocessor. This microprocessor has the ability of tuning the clock speed from 59 to 206 MHz and the core supply voltage from 0.9 to 2.0 V [84]. When the workload is increased in object tracking application, the supply voltage is also increased to fulfill the computation requirement. As can be seen in Fig. 3.4, the results showed that energy consumption is reduced by approximately up to 60% without performance degradation.

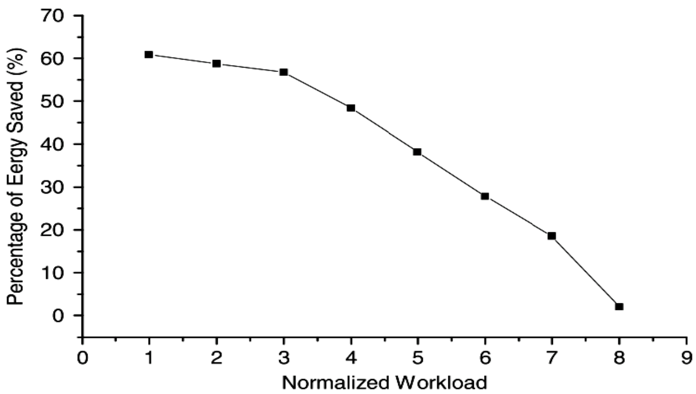


Fig. 3.4 Percentages of the saved energy in terms of the workload [84]

DTM algorithms are developed to control CPU energy. DTM has been proposed as a type of micro-architectural solutions and software strategies to gain the highest processor performance under a peak temperature limit. However, it can be used to slow-down or shutdown the electronic devices when the chip approaches its thermal limit. Jung in [32] proposed a predictive approach based on constructing and utilizing a continuous-time Markovian decision process model of the microprocessor chip to control the temperature of the hardware. This model can be applied for a sensor node to switch the devices to sleep mode when the temperature exceeds its limit consequently achieving reduction in the consumed power.

Finally, through DMS the modulation level is adjusted according to the instantaneous communication load. In the transceiver, the energy is consumed into two components [84]:

- **The electronic circuitry:** including the frequency synthesizer, phase-locked loop, and filtering circuits. Here, the power consumption is independent on the modulation level and the amount of consumed power is constant once the modulation level is fixed.

- **The RF power amplifier:** the choice of the modulation level M is highly affect the total energy consumption and the communication latency. For instance, if parameters such as the energy consumed to start-up transmitter circuitry ($E_{start}(M)$), the power consumed by the electronic circuitry ($P_{elec}(M)$), and the power consumed by output radiation amplifier under the modulation level M ($P_{RF}(M)$) are given. Therefore, the energy cost for transmitting one information bit can be determined as follow [14]:

$$\boldsymbol{E_{bit}} = \frac{\boldsymbol{E_{start}}}{\boldsymbol{L}} + \frac{\boldsymbol{P_{elec}(M)+P_{RF}(M)}}{\boldsymbol{R_s \times log_2(M)}} \times (\boldsymbol{1} + \frac{\boldsymbol{H}}{\boldsymbol{L}}) \tag{3.1}$$

With H and L are packet payload size, and, respectively, and Rs is the symbol rate for the M-ary modulation scheme. From equation 3.1, it is shown that the energy consumption is highly depending on the modulation level. In addition, there is a tradeoff between the bit error rate and reducing the transmission power. Once the modulation level increases to reduce $P_{elec}(M)$, it is necessary to increase the transmission power $P_{RF}(M)$ to keep the bit error rate (BER) within the acceptable level. Therefore, it is mandatory to determine the optimum value for M during low and high workload.

3.2.2.2. Duty Cycling Techniques

As stated in section 3.1, sensor nodes are deployed randomly where a number of nodes remain inactive for long periods. Therefore, their transceivers are not transmitting or receiving.

In order to minimize the power consumed during idle listening, it is necessary to switch the transceiver to the Sleep mode as soon as there is no more data to send/receive, and should be resumed as soon as a new data packet becomes ready. Thus, the wasted power in the active mode is reduced [95]. Consequently, sensor nodes alternate between active and sleep periods depending on network activity. This behavior is usually referred to as duty cycling [27]. However, each node is a member in a network and its function is to cooperate with other nodes to satisfy the application requirement. Thus, the on/off behavior of a node has to be scheduled with other nodes in the network. Consequently, a sleep/wakeup scheduling algorithms based on which sensor nodes decide when to transition from active to sleep and vice versa have to be utilized. Thus, the distributed algorithms have to adaptively select a subset of sensor nodes to be active while other nodes switch to sleep mode.

The process of choosing the optimum subset is called Topology Control (TC). In addition, if the concept of duty cycling is applied to the active subset, then they will switch between idle and sleep modes to conserve energy. This switching behavior is called Power Management (PM).

Now, a number of distributed protocols for PM and TC that exploit the concept of duty cycling for maximizing the network lifetime will be addressed. TC protocols are used to determine the nodes that will undergo in sleep mode as well as the periods of this sleeping. Many protocols are considered to determine these requirements [71]:

- **The location-based protocols:** in this approach, the TC policy is adapted in accordance with the exact location of the sensor node. For instance, Ya Xu el al, proposed in [104] geographical adaptive fidelity technique (GAF) in which the sensing area is divided into virtual grids as shown in Fig. 3.5. The sensor nodes in grid A are able to communicate with the nodes in grid B. Therefore, the nodes in each grid have to elect one node to complete the routing strategy while other nodes undergo in sleep mode. Afterwards, the election is repeated after period to select other representatives of these grids. This method is independent of the operational routing protocol consequently; it can be used along with any existing protocol.

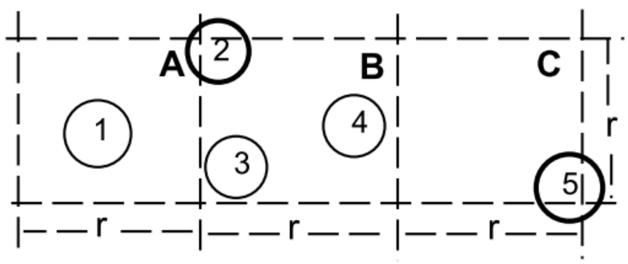


Fig. 3.5 Virtual grid in GAF

Ayad Salhieh et al proposed a power-aware routing protocol, called Directional Source-Aware Protocol (DSAP) [1]. The main idea of this protocol is to reduce the total power through adapting routes to the available power. In DSAP, each sensor node is assigned an identifier that locates the node with respect to the four edges of the network. As can be seen in Fig. 3.6, node 31 for instance has identifier of (1, 3, 4, 2). This means that there is one node to the edge in the left, three nodes upward, four nodes downward, and two nodes in the right direction. Afterwards, all identifiers are exchanged among the sensor nodes consequently; each node contains the identifiers of other nodes. Therefore, when transmitting a message, the destination node identifier is subtracted from the source node identifier. At most two positive numbers are resulted describing in which way the message needs to move, one in either north or south, and one in either east or west while negative numbers are disregarded.

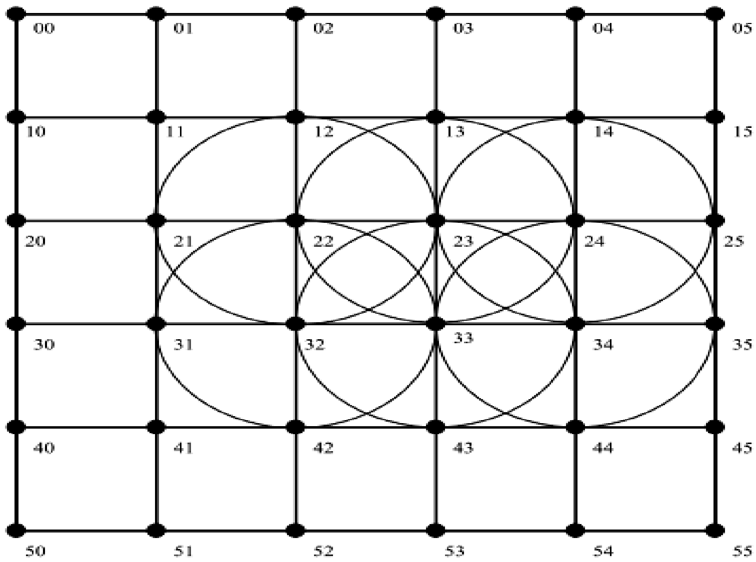


Fig. 3.6 2D Topology with up to four neighbors

M. Zorzi et al [59] proposed a forwarding technique, called Geographic Random Forwarding (GeRaF). The technique is based on geographical location of the nodes involved and random selection of the relaying node via contention among receivers. The authors assumed that the locations of the sensor node and the destination node are known to the node itself. Thereby, when a node has a packet to send, it broadcasts its message as well as its own address and the address of the intended destination. All active nodes will receive the packets and assess their own priority in trying to act as a relay, based on how close they are to the destination. As a result, packets are geographically routed without any routing tables or topological information except for the location of the destination. Nodes in the network are randomly switched between active and passive modes. Therefore, the closest node to the destination will be the best choice.

- **Connectivity-driven protocols:** Alberto et al [2] proposed an Adaptive Self-Configuring Sensor Networks Topologies (ASCENT) for controlling the network topology. In this approach, the decision of joining the routing strategy or continuing in the sleep mode is taken based on information about connectivity and packet loss that are measured locally by the node itself. Nodes in sleep mode are listening to the traffic without transmitting packets until the sink node detects high message loss from sources. Afterwards, the sink node asks passive nodes to switch to the active mode. However, they only collect information about the network status without interfering with other nodes. Their mission starts when the active neighbors are depleted from energy.

C. Schurgers et al proposed in [13] a new technique, called Sparse Topology and Energy Management (STEM) to establish communications in the presence of sleeping nodes. In this protocol, each passive nodes wakes up periodically to listen to the traffic. However, they send out beacons polling a specific user if they need to communicate with this user. The communication between the two nodes starts when the polled node wakes up and receives the poll.

- **Transmission power control:** many research studies investigated the relationship between transmit power and connectivity. Choosing the optimum transmission ranges and the suitable number of neighbors that minimize the power consumption were discussed. For instance, the network can be modeled as a geometric random graph and an analytical expression may be derived that allows the determination of the required transmission range r. Consequently, the optimum number of nodes to cover a certain area can be determined.

P. Santi et al proposed a method for energy saving by means of a probabilistic approaches [71]. They showed that if the probability of connectedness is sufficient, the transmitting range r of the sensor nodes could be reduced substantially from the deterministic requirements. In addition, they proposed an optimum value of r in the two and three dimensions.

R. Wattenhofer et al [75] proposed a distributed algorithm where each node makes local decisions about its transmission power and these local decisions collectively guarantee global connectivity. In this algorithm, each node transmits a discovery message to its neighbors with arbitrary power p where the value is ranged from zero to P. Each receiving node responds with an acknowledgment message. Thereby, each node records all acknowledgements and the direction they came. They determine the direction by IEEE antenna. More neighbors are discovered by increasing the transmission power until at least one neighbor node in every direction or until it reaches the maximum transmission power P. consequently, the routing is made through nodes that consume minimum power for the transmission.

- **Schedule-based protocols:** in these techniques, scheduling methods are employed to manage the duty cycling periods. For instance, Time Division Multiple Access (TDMA) may be used to assign a time slot for each node to transmit its information within while other times the nodes undergo in sleep mode.

W. Heinzelman et al proposed Low-Energy Adaptive Clustering Hierarchy protocol (LEACH) [96]. This approach tries to distribute the energy load among the network nodes by randomly rotating the cluster head among the sensors. LEACH divides the networks into clusters and in each cluster a dedicated node, the cluster head, is responsible for creating and maintaining a TDMA schedule. Other cluster members switch to sleep mode all times except during its time slot. The cluster head aggregates the data of its members and transmits it to the sink node or to other nodes for further relaying. However, cluster heads perform additional functions given that they have limitation in the allocated energy. Then, cluster heads will lose its energy and die quickly. In LEACH, the nodes inside each cluster elect their head based on information such as when this candidate node served as cluster head the last time where node that has not been a cluster head for a long time is more likely to elect itself than a node serving just recently.

3.2.2.3. Data Driven Approaches

The authors of [27] classified the data driven techniques into in-network processing, data compression, and data prediction techniques. In this section, we will focus only on the prediction techniques, as they are the basis of our work. As can be seen in Fig. 3.7, there are three types of data prediction algorithms: stochastic, time series forecasting, and algorithmic approaches.

However, time series forecasting will be discussed in the next chapter. On the other hand, stochastic algorithms are based on probabilistic models such as correlation [18], Kalman filter [4] and Dynamic Probabilistic Model (DPM) [6]. In general, temporal correlation in the measured values is an advantage since the next instance value can be predicted using previous instance values with the aid of suitable prediction algorithms.

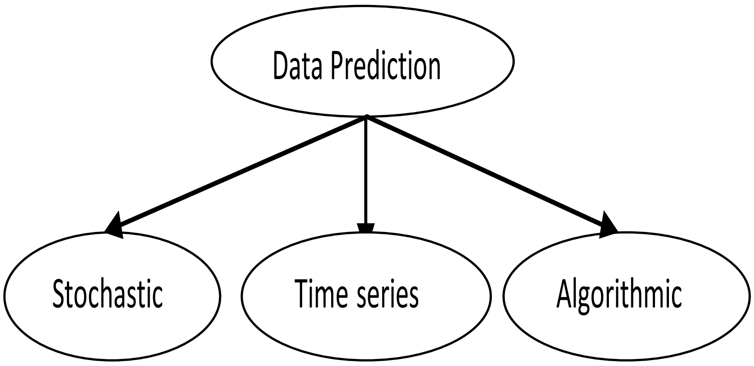


Fig. 3.7 Classification of data prediction algorithms

- **Stochastic Methods:** in these approaches, data are mapped into a random process expressed in terms of a Probability Density Function (PDF). By combining the computed PDFs with the observed samples, the predicted values can be obtained. On the other hand, a state space representation of the phenomenon can be derived, so that forthcoming samples can be guessed by filtering out a non-predictable component modeled as noise.

David et al [18] proposed a technique called KEN; uses replicated dynamic probabilistic models for minimizing communication from sensor nodes to the sink node. Initially, two identical dynamic probabilistic models are implemented in both the sensor nodes and the sink nodes. The sink node and sensor nodes simply compute the expected values of the sensors attributes according to the probabilistic model. However, sensor nodes always possess the ground truth, and whenever they sense anomalous data, they route the data back toward the sink node. As data is routed toward the sink node, spatial correlations among the reported data are used to further lower communication.

- **Algorithmic approaches:** In this approach, it is found that the prediction techniques rely on a heuristic or a state-transition model describing the sensed phenomenon. In classic data gathering techniques, sensor nodes send updates to the sink node at a fixed periodicity.

These methods did not provide the optimum solution for providing high data quality and minimizing the power consumption due to the instability of sensor readings as well as the varying application needs that impose different quality requirements across sensors. Therefore, Q. Han, et al [74] proposed an adaptive data collection protocols for sensor environments that adjusts to these variations while at the same time optimizing the energy consumption of sensors. The proposed algorithm relies on assigning an upper and lower bounds in each node for the measurements. These bounds are sent to the sink, which stores them for each sensor in the network. At the time of acquiring the data, sensors check the samples against the current bounds. If they fall outside the predefined bounds, nodes send an update to the sink.

P. Godfrey et al [66] proposed randomized topology management scheme, called “Naps”. In this scheme, sensor nodes send only a periodic heartbeat message between waking neighbors in order to minimize the transmission for energy conservation.

S. Goel et al [81] proposed a new scheme Prediction-based monitoring for energy-efficient monitoring (PREMON). The authors showed that there are similarities between PREMON and the algorithm followed by MPEG encoder. Therefore, they exploit these similarities for employing the algorithms and theory of MPEG in WSNs. This method exploits the spatial and temporal correlation of the sensors measurements. For instance, they visualize sensor networks as an image with its pixel represents the nodes' measurements. Therefore, the sink node can collect these data and perform correlations. PREMON takes into account the change of the measurements over time. Therefore, it visualizes the network as "a movie" that contains the entire measurements. The sink node receives this movie for a period then executes spatial and temporal correlation and generates the prediction. Afterwards, it broadcasts this prediction to sensor nodes, which in turn compares the actual measurements with the prediction and sends updates to the sink if the accuracy falls below the acceptable level.

S. Goel et al [82] tried to improve the performance of PREMON by using distributed algorithm rather than centralized one as exist in PREMON. In this approach, sensor networks are divided into number of clusters. Each cluster elects its cluster head that performs the role of aggregating node. Members of the cluster collaborate to rotate the responsibility of being the representative to spread the consumption of energy uniformly over the group members. This cluster head is responsible for monitoring and query processing, while other nodes can alternatively switch between idle and sleep modes. The communication between the cluster members and their head is made based on the PREMON algorithm or the default mode. Each node decides whether to use default or PREMON mode based on an estimate of the energy cost associated to the specific operational mode.

Chapter 4

Reliable and Efficient Data Reduction Proposed Techniques

4.1. Introduction

As mentioned previously, one of the basic problems in WSN is the network lifetime. Network lifetime can be defined as the interval of time, starting with the first transmission in the wireless network and ending when the percentage of nodes that have not terminated their residual energy falls below a specific threshold. Such threshold is set according to the type of application (it can be either 100% or less) [94].

Computation of node's lifetime requires knowledge of the time spent in the various states including transmission, reception, listening, and sleeping. It is well known that the energy cost of transmitting 1 Kb of information a distance of 100m is approximately the same as executing three million instructions per second [78]. For this reason, energy efficient models have to be employed reducing the wasteful power that is consumed in the radio communication.

As a conclusion, the transceiver is the device that is responsible for the consumption of sensors' most energy. Therefore, data communication is very expensive in terms of power consumption. It is therefore mandatory to minimize the data items that need to be transmitted to the base station. Based on our experience, sudden changes in sensors' readings are not a common feature of WSN; therefore, utilizing this feature might increase the overall lifetime of the WSN. Some of the proposals in this regard forces the sensors to send only the abrupt changes based on a threshold value. However, the data reliability, in this case, depends on the threshold value defined by the WSN user.

In this chapter, the proposed solutions for the power conservation problem, called Reliable and Efficient Data Reduction Technique (REDR) are addressed. The impact of applying the data reduction techniques on the lifetime of WSNs is investigated. The approaches utilized single and double exponential predictors, fuzzy logic algorithm, and threshold and tolerance e approaches to reduce the number of messages transmitted between the nodes and their neighbors as well as between the nodes and the sink node.

4.2. Description of Data Reduction Technique

One way to maximize the network lifetime and minimize the power consumption is to reduce the number of messages to be sent by sensors to the sink node with guaranteeing some data accuracy. This algorithm, called Dual Prediction Scheme (DPS), achieves data reduction by nominating a subset of sensor measurements and sending only this information to the sink node. Selection must be performed in a way such that the original observation data can be reconstructed within some user-defined accuracy. Reconstruction is most often carried out using prediction algorithms. This can be mathematically formulated as follows:

$$min \sum_{t=0}^{L} \sum_{s=1}^{n} M_{st} X_{st} \qquad \forall_{s,t} \tag{4.1}$$

Provided that:

$$L_{st} \geq \mu \qquad \forall_{s} \tag{4.2}$$

$$\left| T_{st} - T_{s(t-1)} \right| \leq \gamma \quad \forall_{s} \tag{4.3}$$

Equation (4.1) tries to minimize the number of messages M_{st} that can be sent by a sensor *s* starting from *s=1* to *n* at any time *t*. At the same time, a sensor s is considered died when its lifetime or residual energy L_{st} is reached a certain threshold μ as stated in equation (4.2). X_{st} is a binary variable that is set to *1* when a message is to be sent and *0* otherwise. On the other hand, it is necessary to maintain the accuracy of the data T_{st} sent by a sensor s at time t high and suitable for the application's requirements as described in equation (3).

However, the data reliability, in this case, depends on the threshold value defined by the WSN user ($\gamma\ in\ our\ case$). The sink node will be working on values other than the original readings of the sensor nodes. Therefore, it is obvious the reliability and data reduction are contradicting terms since the reliability requires sending every piece of information to the sink node while the data reduction and energy conservation require the minimization of the number of messages that are received by the sink node for other nodes. Our proposal in this thesis considers both the data reliability as well as the data reduction for maximizing the overall network lifetime.

The main idea behind utilizing these predictors in sensors' data reduction is to use typical predictors on sensor nodes as well as on the sink node as shown in Fig. 4.1. In a clustered network, the same predictor will be running on the cluster heads as well. Predictors running on sensors are used to predict the sensor's readings, where the predictor on the sink node is initialized after the sensor node sends a number of readings to it. Therefore, on each sensor, we have two values, the sensor's reading, and the predicted value. If the difference between these two values, which is called the error signal, is lower than a certain threshold value then the transmission between the two nodes is canceled as given in Fig. 4.1, and the sink node stores the predicted value generated from its own predictor as the sensor reading at this time. On the other hand, if the error signal is greater than this threshold, then the sensor reading is transmitted to the sink node, and the sink node uses this value to update its predictor. The steps of the DPS inside the sink and source nodes are demonstrated in the flowcharts shown in Figs. 4.2 and 4.3.

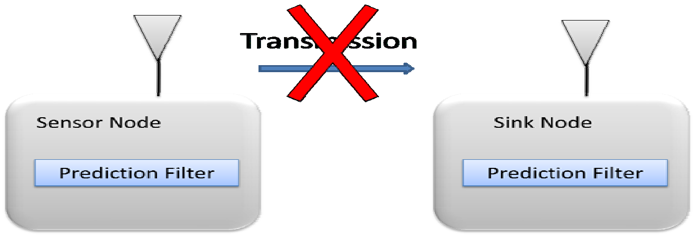


Fig. 4.1 Dual Prediction Scheme

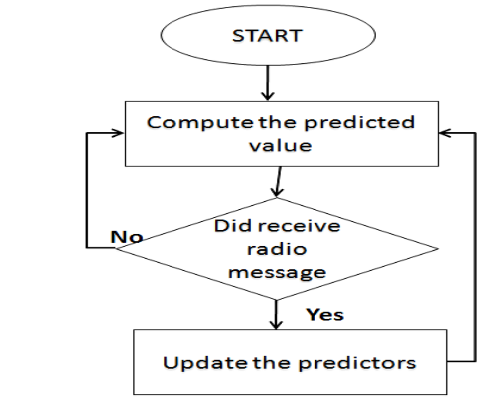


Fig. 4.2 Flowchart of the operations inside the sink node

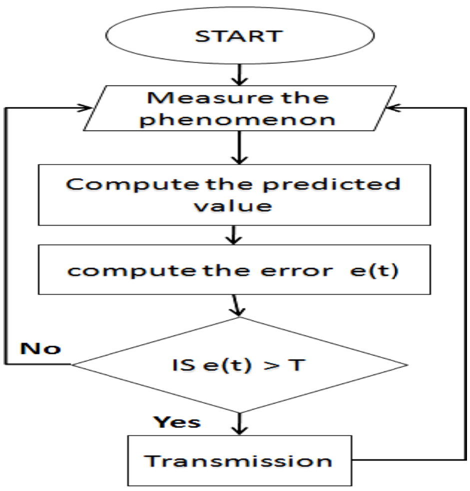


Fig. 4.3 Flowchart for the operations inside the source node

Due to the importance of data reduction techniques, there are many techniques have been proposed. Santini, et al [87] proposed data reduction technique that uses Least Mean Square (LMS) adaptive algorithm. Nicholas

Paul et al implemented this algorithm on FPGA kit where they showed that this method managed to increase the network lifetime by 18,962.5 % when compared to an always-on solution [60]. However, sensors' data usually have a trend and might seasonal information that we can benefit from. Moving Average (MA) and Auto-Regressive prediction algorithms are also used in WSN.

There are many prediction schemes such as Adaptive filters based predictors, Markov predictors, Linear Extrapolation, Neural network based predictors, and Exponential smoothing predictors. However, the data reduction technique based on the exponential smoothing algorithms is investigated since they are more appropriate if the data include rapid fluctuation. In addition, they need very small set of data to predict the incoming values, so they require small memory footprint. In addition, they provide low computation overhead. Moreover, another computational intelligence technique is employed by utilizing the Fuzzy Logic in data reduction.

4.3. Exponential Smoothing Predictors

The predictors are required to be accurate and simple. Therefore, the predictors are implemented through using the concept of Exponential Smoothing (ES) [22]. ES has gained popularity mostly because of its usefulness as a prediction tool. For instance, Makridakis *et al*. [92], ranked SES as the best choice for one-period-ahead forecasting, from among 24 other time series methods and using a variety of accuracy measures. Thereby, SES often will produce quite accurate forecasts.

ES is a mathematical-statistical method of prediction used in industrial engineering where the prediction is based on detecting significant changes in data. This algorithm is very popular scheme to produce a smoothed Time Series. It assigns exponentially decreasing weights, as the observation gets older as shown in Fig. 4.4.

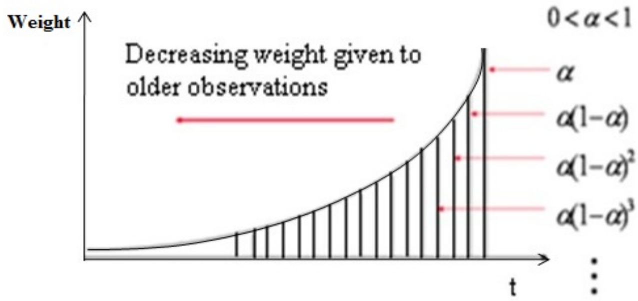


Fig 4.4 Weights decay exponentially as observations get older [73]

In other words, recent observations are given relatively more weight in forecasting than the older observations. The reason for this is that the future may be more dependent upon the recent past than on the distant past. Single Exponential Smoothing (SES) is used for stationary data whereas Double Exponential Smoothing (DES) is better at handling trends. Triple Exponential Smoothing (TES) is better at handling parabola trends.

It is commonly applied to financial market and economic data, but it can be used with any discrete set of repeated measurements. Time series prediction is one type of the prediction techniques that heavily used in many applications including as inventory control, tracking, and other Applications in Finance. Also, it may be used for saving the handover' latency in WiMAX applications as proposed in [85]. Data reduction is achieved by implementing identical predictors on sensor nodes as well as on the sink node. In a clustered network, the same predictor will be running on the cluster heads as well.

However, some terminologies have to be defined prior to the commencing in describing the different types of exponential smoothing.

Time-Series data are data, which are measured over time. In most applications, the period among measurements is uniform. In addition, any time series is composed of four components:

- A *linear trend* is any long-term increase or decrease in a time series in which the rate of change is relatively constant.
- A *seasonal component* is a pattern that is repeated over the time series and has a recurrence period of at most one year.
- A *cyclical component* is a pattern within the time series that repeats itself throughout the time series and has a recurrence period of more than one year.
- The *random component* is the changes in the data that are unpredictable and cannot be associated with the trend, seasonal, or cyclical components.

Each sensor on a node produces a time series representing the development of the sensed physical variable over space and time. The exponential smoothing methods are classified according to the nature of the time series. For instance, SES provides reasonably low performance when the history data incorporates a trend or seasonality components. Therefore, performance is improved by the aid of DES or TES. In the following sections, the three types of ES will be discussed in more details.

4.3.1. Single Exponential Smoothing

Robert G. Brown [20] proposed this idea in 1944 while he was working for the US Navy as an Operations Research analyst. This method is used when the data has a *mean* that is either stationary or changes slowly with time. In other words, for higher prediction accuracy SES is used when the data does not include trend or seasonality components. The model of SES predictor is expressed by the following formulae [22].

$$F_{t+1} = \alpha\, y_t + (1-\alpha) F_t \tag{4.4}$$

Where F_{t+1} is the prediction for the next period, α is the smoothing constant, y_t is the measured value in period t, and F_t is the old forecast for period t. If the smoothing equation is recursively applied to F_{t+1}, we get:

$$F_{t+1} = \alpha\, y_t + \alpha(1-\alpha) y_{t-1} + \alpha (1-\alpha)^2 y_{t-2} + \dots + (1-\alpha)^{n+1} F_{t-n} \tag{4.5}$$

As time passes, the smoothed statistic F_t becomes the weighted average of a greater and greater number of the past observations y_{t-n}, and the weights assigned to previous observations are in general proportional to the terms of the geometric progression $\{1, (1-\alpha), (1-\alpha)^2, (1-\alpha)^3, \dots\}$ as shown in Fig. 4.4.

The smoothing constant must satisfy the following inequality $0 < \alpha < 1$, if α is chosen close to 1, it will have less of a smoothing effect and give greater weight to recent changes in the data, while values of α closer to zero have a greater smoothing effect and are less responsive to recent changes. For higher accuracy, the value of α which satisfies the smallest Mean Squared Error (MSE) is chosen for use in producing the future predictions. The initial value of F_t plays an important role in computing all the subsequent values. One option is to assign the first observation as an initial value. However, more accurate initialization can be obtained by averaging the first five observations.

4.3.2. Double Exponential Smoothing

During the 1950s, Charles C. Holt [20] developed a method for exponential smoothing of additive trends, Known as Holt's method. the predictor's operation is initiated by decomposing the data under consideration into level and trend signals afterwards these signals are smoothed using equations (4.6) and (4.7) while the predicted value are found using equation (4.8).

$$L_t = \alpha y_t + (1-\alpha)(L_{t-1} - b_{t-1}) \tag{4.6}$$

$$b_t = \beta(L_t - L_{t-1}) + (1-\beta) b_{t-1} \tag{4.7}$$

$$F_t = L_t + m b_t \tag{4.8}$$

- L_t is the estimate of the level at time t,
- y_t is the actual value of series in period t,
- α is the data smoothing factor, $0 < \alpha < 1$,
- β is the trend smoothing factor, $0 < \beta < 1$,
- b_t is the estimate of the slope of the series at time t,
- m is the periods to be predict into the future.

In equation (4.6) L_t is adjusted directly for the trend of the previous period, b_{t-1}, by adding it to the last smoothed value, L_{t-1}. This helps to eliminate the lag and brings L_t to the appropriate base of the current value. Afterwards, equation (4.7) evaluates the trend at time t, which is expressed as the difference between the last two values. The equation is similar to equation (4.4), but here applied to the updating of the trend. The smoothing constants α and β are chosen independently between 0 and 1, where their values must satisfy the smaller MSE. To start the process, both L_1 and b_1 must be initialized by setting $L_1 = y_1$ and $b_1 = y_2 - y_1$. Thus, no forecasts can be made until y_1 and y_2 have been observed. By convention, we let $F_1 = y_1$.

4.3.3. Triple Exponential Smoothing

Holt developed a method for smoothing seasonal data in 1960, Winters tested Holt's methods with empirical data, and they became known as the Holt–Winters forecasting system. The ideas behind smoothing with trend and seasonality is to "De-trend" and "De-seasonalize" time-series by separating *base* from *trend* and *seasonality* effects. In other words, in equation (4.9), the base is smoothes in usual manner like equation (4.4) using smoothing constant α while in equation (4.10) the trend is also smoothes using smoothing constant β and finally the seasonality component in equation (4.11) is smoothes using smoothing constant γ. Afterwards, the prediction is computed in equation (4.12).

$$L_t = \alpha y_t + (1-\alpha)(L_{t-1} + T_{t-1}) \tag{4.9}$$

$$T_t = \beta(L_t - L_{t-1}) + (1-\beta)T_{t-1} \tag{4.10}$$

$$S_t = \gamma \frac{y_t}{L_t} + (1-\gamma)S_{t-m} \tag{4.11}$$

$$F_{t+k} = (L_{t-1} + kT_{t-1})S_{t+k-m} \tag{4.12}$$

The smoothing constants are optimized in order to achieve the smallest MSE. In order to determine initial estimates of the seasonal indices we need to use at least one complete season's data (i.e. s periods).Therefore, trend and level are initialized at period s. the level, trend, and seasonal index initializations are expressed as follow:

$$L_s = \frac{1}{s}(y_1 + y_2 + \cdots y_s) \tag{4.13}$$

$$T_s = \frac{1}{s}\left(\frac{y_{s+1} - y_1}{s} + \frac{y_{s+2} - y_2}{s} + \cdots + \frac{y_{s+s} - y_s}{s}\right) \tag{4.14}$$

$$S_1 = \frac{y_1}{L_s}, S_2 = \frac{y_2}{L_s}, \cdots, S_s = \frac{y_s}{L_s} \qquad (4.15)$$

In this thesis, the mathematical model of TES will not be employed since it is only suitable for data stream with trend and seasonality. The research in this work is limited to the previous two prediction models since the real data that we have has no seasonal features. However, the Holt-Winter's method will be studied when sensors' seasonal data are available.

4.4. Fuzzy Logic

Lotfi A. Zadeh proposed the concept of fuzzy logic in 1965 [89] , and presented it not as a control methodology, but as a way of processing data by allowing partial set membership rather than crisp set membership or non-membership.

Fuzzy logic is a multi-valued logic, where it formalizes reasoning when dealing with vague terms. The decisions are not limited to either true or false, or as with Boolean logic either *0* or *1*. Therefore, fuzzy logic algorithms take into consideration the degrees of truthfulness and falsehoods. For instance, the Boolean logic is not only *0* and *1* but also all the numbers that fall in between.

In order to understand the concept of fuzzy logic, the following example is considered. As can be seen in Fig. 4.8, each circle represents a group. In each group, as closer to the center, the membership in that group is "stronger". On the other hand, as farther from the center, reduction in this membership occurs because of growing the other membership from the second group. Consequently, it is true that, a valid value may be member of Group 1, Group 2, or both of them.

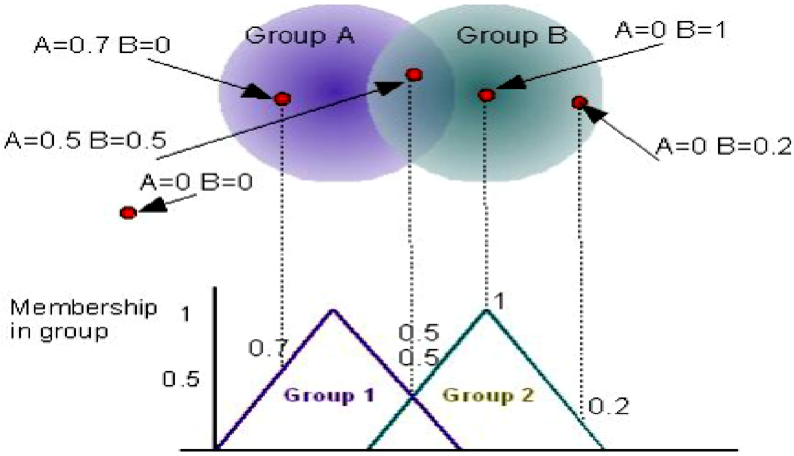


Fig. 4.5 Fuzzy groups

The first application of fuzzy logic was in the area of fuzzy controllers. Japanese applied the algorithms in the control systems for subway trains, and after the success of these systems, they extend the applications to other areas such as elevator control systems and air conditioning systems. In the early 1990s, Japanese applied fuzzy controller in consumer products, such as camcorders, washing machines, vacuum cleaners, and cars [7].

Nowadays, the concept of Fuzzy Logic is employed widely in many systems including automobile engine, automatic gear control systems, air conditioners, automatic focus control, video enhancement in TV sets, Pattern Recognition (Image Processing, Machine Vision), washing machines, behavior-based mobile robots, sorting and handling data, Information Systems (DBMS, Information Retrieval) [5]. Moreover, fuzzy logic can be used in the implementation of the predicative fuzzy-logic controller for automatic operation of trains, laboratory water level controllers, a navigation system for automatic cars, decision support, graphics controllers for automated police sketchers, traffic control systems, and more. In addition, the benefits of fuzzy logic can be exploiting in the expert systems such as decision-support systems, financial planners, diagnostic systems for determining soybean pathology, and a meteorological expert system in China for determining areas in which to establish rubber tree orchards.

The components and general architecture of fuzzy logic system is shown in Fig. 4.6. The process of fuzzy logic is explained as follow:

1. Firstly, crisp set of input data are gathered and converted to a fuzzy set using fuzzy linguistic variables, fuzzy linguistic terms, and membership functions. This step is known as Fuzzification.
2. An inference is made based on a set of rules.
3. The results from each rule are aggregated into one fuzzy variable.
4. Finally, the resulting fuzzy output is mapped to a crisp output using the membership functions in the Defuzzification step as shown in Fig. 4.6.

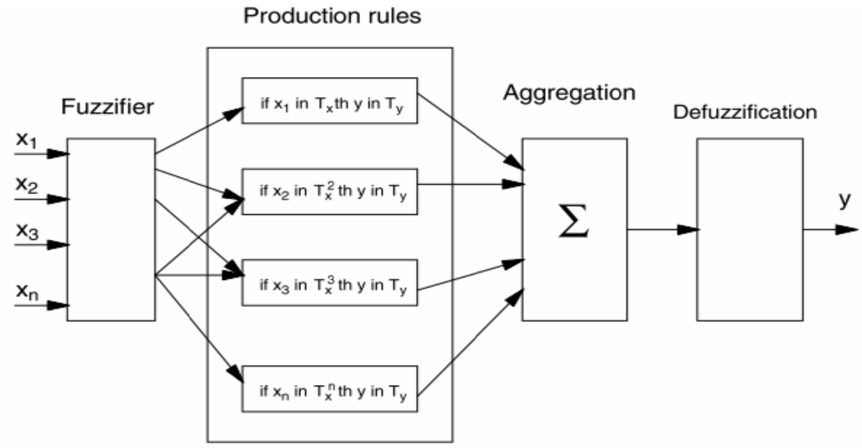


Fig. 4.6 Fuzzy logic System [89]

However, some terminologies that are used in the fuzzy logic algorithm have to be defined:

4.4.1. Linguistic Variables

Linguistic variables are a representation for the system's input or output variables where they convert the numerical values into words or sentences from a natural language. A linguistic variable is decomposed into a set of linguistic terms form the normal speech. For instance, in forest fire detection application using sensor nodes, temperature is the input linguistic variable, which represents the temperature of the forest. To qualify the temperature, terms such as "hot" and "cold" are used in real life. These are the linguistic values of the temperature. Then, $T(t)$ = {too-cold, cold, warm, hot, too-hot} can be the set of decompositions for the linguistic variable temperature. Each member in these set is called a linguistic term and can cover a portion of the overall values of the temperature.

4.4.2. Membership Functions

These functions are used to map crisp input values to fuzzy linguistic terms and vice versa. For example, membership functions for the linguistic terms of temperature variable are shown in Fig. 4.7. According to the definition of fuzzy logic, a numerical input value does not have to be fuzzified using only one membership function. Thereby, a value can belong to multiple sets at the same time. For example, according to Figure 4.7, a temperature value can be considered as "cool" and "warm" at the same time, with different degree of memberships.

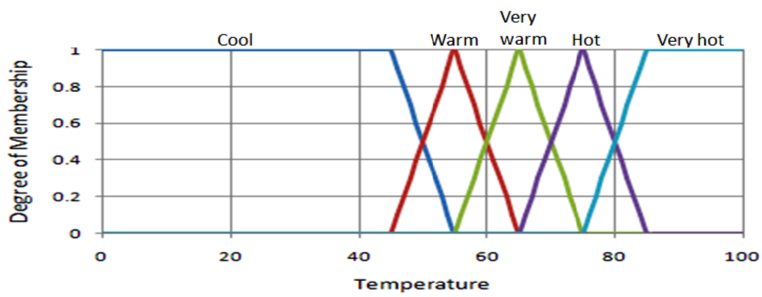


Fig. 4.7 Membership function for temperature

4.4.3. Fuzzy Rules

As show in Fig. 4.6, the fuzzy output variables are obtained through evaluating a number of rules. These rules are based on a simple IF-THEN statement with a condition and a conclusion [11]. For instance, Table 4.1

contains a sample of the rules for controlling the breaks pressure for two consequence cars according to the speed of these two cars as well as the distance between them. To evaluate the rules, fuzzy set operations are considered in which AND function is mapped to the minimum of the degrees of membership. While, OR function is mapped to the maximum of the degrees of membership.

Table 4.1 Sample of fuzzy rules

1. IF (Distance is Low) AND (Speed is High) THEN Breaks Pressure (High).
2. IF (Distance is Medium) AND (Speed is Medium) THEN Breaks Pressure (Medium).
3. IF (Distance is Low) AND (Speed is High) THEN Breaks Pressure (High).
4. IF (Distance is Low) AND (Speed is Low) THEN Breaks Pressure (Medium).
5. IF (Distance is High) AND (Speed is Low) THEN Breaks Pressure (Low).

4.4.4. Fuzzy Inference

In this stage, the outputs from the fuzzy rules are aggregated to start the defuzzification phase. However, there might also be few rules producing different strength values for the same conclusion. For instance, in the last example, the output linguistic variable "Medium" many be fired from many rules with multiple degrees of membership [X_1, X_2, X_3]. Therefore, a single cumulative strength value should be evaluated before defuzzification. In some cases, the fuzzy OR function may be used (select the maximum strength output for each conclusion), but the most common and precise method is the Root Sum Square (RSS). The RSS method is expressed mathematically as follow:

$$X = \sqrt{(X_1 + X_2 + X_3)} \tag{4.16}$$

With X is the degree of membership for the output variable “Medium”. These values would be applied to the output membership function for defuzzification.

4.4.5. Defuzzification

In this phase, the fuzzy output variables are transformed into crisp output value. Firstly, all the fuzzy conclusions obtained by inference are aggregated into a single conclusion. Defuzzification is performed according to the membership function of the output variable [35]. Many methods are used for the defuzzification including height-center of area method, max criterion method, first of maxima method, and Middle of maxima method. However, the mostly used algorithm is the center of gravity for singleton method in which the output value is obtained as follow:

$$U = \frac{\sum_{i=1}^{p}[u_i \mu_i]}{\sum_{i=1}^{p} \mu_i} \tag{4.17}$$

With U is the result of defuzzification, u is the output variable, p is the number of singletons, μ is membership function after accumulation, and i is the index. On the other hand, more details about the other defuzzification methods can be in [35].

4.4.6. Fuzzy based power conservation method

In the multi-modal WSN, multiple environmental phenomena have to be collected and sent to the end user for evaluation; however, this will accelerate the depletion of the sensor node's energy. Fuzzy logic is a proposed solution for this problem through minimizing the transmissions in the network. Hence, adding intelligence to the nodes will convert their function from sending measurements to sending information that is generated based on the measured parameters. In this work, the input to the fuzzy logic algorithm is values of temperature, humidity, and light intensity at a certain area. These values will be transformed to linguistic variables in order to be processed using the fuzzy rules. The output of the fuzzy logic algorithm is a value that represents the probability of certain event to occur. Therefore, the sensor nodes will not turn on its transceiver unless this probability exceeds a threshold set by the user.

4.5. Threshold and Tolerance Based Approaches

In this section, REDR is implemented based on employing thresholds. In other words, a certain value called threshold is used. When the measurements are below threshold, the nodes will turn off its transceiver. However, if this threshold is exceeded, the nodes send a message to the base station.

Similar performance for the REDR can be obtained through such technique that employs the difference between the present and past readings in order to reduce the radio messages. In this case, nodes undergo in the sleep mode as long as this difference is lower than a certain tolerance defined by the user. In order to verify all the proposed solutions, simulations and practical experiments will be provided in the next chapter.

Chapter 5

Experimental and Simulation Results

5.1. Introduction

In this chapter, the performance of the proposed approaches for maximizing the WSN lifetime is tested through simulations and practical experiments. Several tests are made on the WSN before and after the implementations of exponential smoothing predictors, fuzzy logic algorithm, and other simple algorithms based on comparing present and past measurements. This chapter is divided into two main sub-sections. In the first one, the simulations of WSN when applying the proposed approaches will be presented. While in the second one, the results of the practical experiments will be shown.

5.2. Simulation Results

A WSN simulator shown in Fig. 5.1 is developed to investigate how our reduction techniques reduce the transmitted data among the sensors as well as between the sensors and the sink node. The simulator developed by David J. Stein [102] is modified. The simulator is written in C# and used to detect and report certain events across certain areas. The author simulated the WSN as a connected graph and sensors always transfer their data towards the sink node. Sensors decisions in sending and/or forwarding messages are based on their residual energy as proposed in [46]. The simulator is modified to cope with our desired mission as follow:

- The event is changed from detecting moving objects to sensing environmental phenomena such as temperature, humidity, or pressure.

- An energy model is added to the simulator to estimate the lifetime accurately.
- The data reduction techniques are implemented to increase the network lifetime.

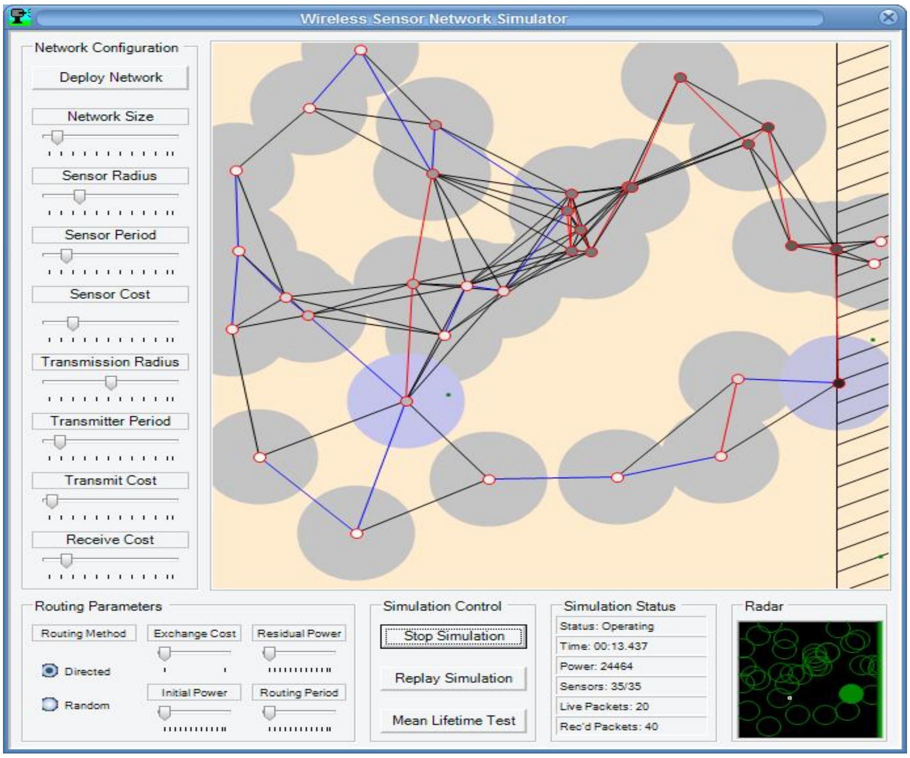


Fig. 5.1 WSN simulator [102]

5.2.1. Energy Model

For the purpose of accuracy, a similar energy model to the one used in [96] is implemented as shown in Fig. 5.2. In this model, it is assumed that the radio channel is symmetric such that the energy required for transmitting a message from node A to node B is the same as the energy required for transmitting a message from node B to node A for a given Signal to Noise Ratio (SNR). In addition, all sensors are assumed to be sensing the environment at a fixed rate and thus always have data to send to the end-user. As given in Fig. 5.2, the first order radio model stated that the energy for transmitting 1 bit data over distance d is:

$$E_t = a_1 + a_2 * d^k \tag{5.1}$$

- a_1 is the energy spent by transmitter electronics (a_1 = 50 nJ/bit),
- a_2 is the transmitting amplifier (a_2 = 100 pJ/bit/m^2),
- k is the propagation loss exponent.

On the other hand, the energy for receiving 1 bit data E_r is equal to the energy spent by receiver electronics (a_3 = 50 nJ/bit). This model is used throughout all of our experiments.

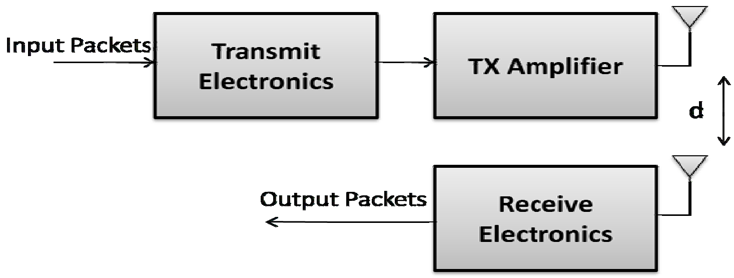


Fig. 5.2 The First order radio model

5.2.2. Simulator Setup

Our simulator runs on a Windows 7 machine with 2.13 GHz processor and 2 GB RAM. 35 nodes are randomly deployed in an area of 460 X 300 cm. It is assumed that nodes are equipped with a 1 cm^3 of non-rechargeable lithium battery (at maximum energy density of 2880 J/cm3 or 800 watt hour per liter) were to consume 100 µW of power on average.

The lifetime of the WSN is considered as the period between the start of simulation process until the depletion of at least one sensor node from its energy. In our experiments, it will not be practically to consider the network lifetime as the running time of the simulator until a node's energy is depleted. Thus, the lifetime is counted based on the number of iterations that the simulator takes until a node dies. The iteration is described as a complete pass over all the nodes having data to be sent and/or forwarded until their data reaches the sink node.

In the following subsections, the influence of the data reduction technique will be tested based on exponential smoothing predictors, fuzzy logic algorithm, and threshold and tolerance based algorithms on the lifetime of WSN using the described simulator.

5.2.3. Data Reduction Based On Exponential Smoothing Predictors

Data reduction proposal based on single and multiple smoothing predictors is test for both single modal and multimodal WSN. However at the beginning, the predictors have to be examined separately to determine the best smoothing constant found in (4.4), (4.6), and (4.7). Therefore, our experiments are categorized into two categories, which are experimental category based on real data and simulation results based on simulation to the WSNs environment. Throughout the next sections, we will elaborate more on these categories.

5.2.3.1. Predictors' parameters

In the first set of experiments, the exponential smoothing predictors were tested on a set of real world data, which is publicly available at [44]. Every 31 seconds, humidity, temperature, light intensity, and voltage values were collected from 54 Mica2Dot sensor nodes [17] that were deployed in the Intel Berkley Research Lab between February 28th and April 5th, 2004 as shown below in Fig. 5.3. As can be seen in Fig. 5.3, the black circles represent the location of the sensor nodes.

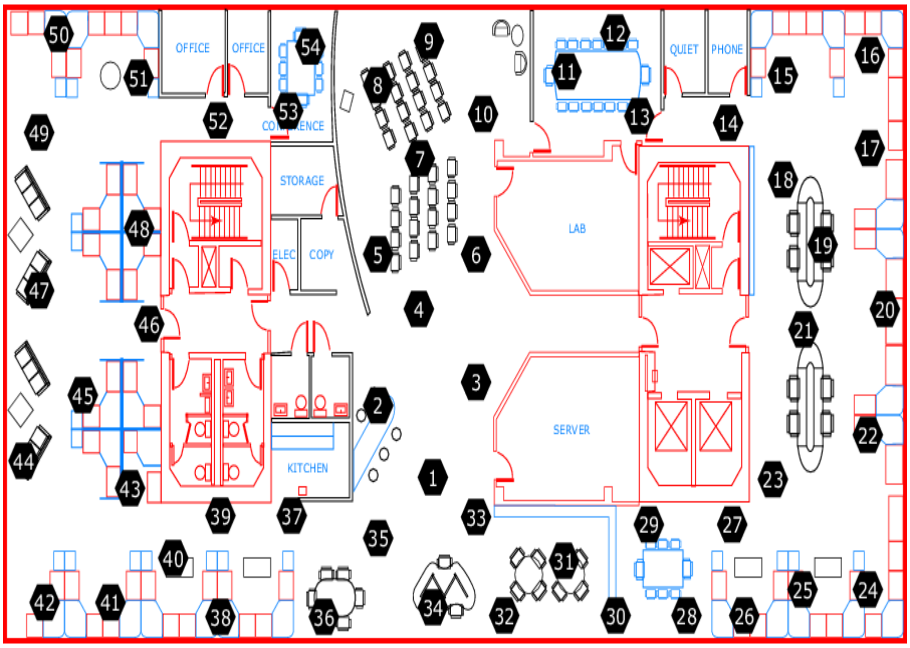


Fig. 5.3 Layout of Intel lab [44]

A file, downloaded from [44], contains a log of about 2.3 million readings collected from the 54 sensor nodes. These measurements are used as input to SES and DES algorithms in order to examine many values for the coefficient constants and find the best values that minimize MSE. As shown in Fig. 5.4, the real data frame consists of:

- The date, time of measurements,
- Epoch number which is a monotonically increasing sequence number from each mote,
- Temperature readings are in degrees Celsius,
- Humidity readings which are temperature corrected relative humidity, ranging from 0-100%,
- Light intensity readings are in Lux (a value of 1 Lux corresponds to moonlight, 400 Lux to a bright office, and 100,000 Lux to full sunlight).

Date	Time	Epoch	Mote ID	Temperature	Humidity	Light Intensity	Voltage

Fig. 5.4 Real data frame

The results given in Figs. 5.5 and 5.6, show that the larger the smoothing constant the lower the Mean Square Error (MSE). This result verifies the fact that the exponential predictors become more accurate when the smoothing constants approach one. In addition, it is noticed in Figs. 5.5 and 5.6 that the MSE in case of temperature values is smaller than that in case of humidity.

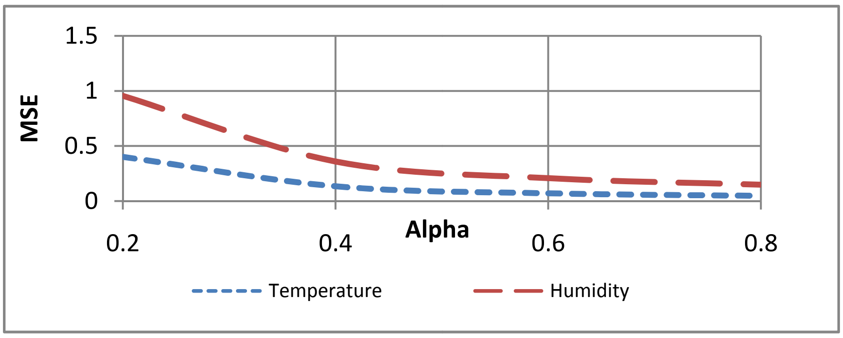


Fig. 5.5 MSE decreases as α increase.

As can be seen in Figs. 5.5 and 5.6, starting from β=0.4 the MSE values are almost the same. Therefore, β and α are selected to be 0.4 and 0.6 respectively. The reason behind choosing α to be 0.6 not 0.8, as given in Fig. 5.5, based on our observation in DES, produces better MSE when α=0.6 not 0.8. Therefore, to standardize the experiments, we set α to 0.6.

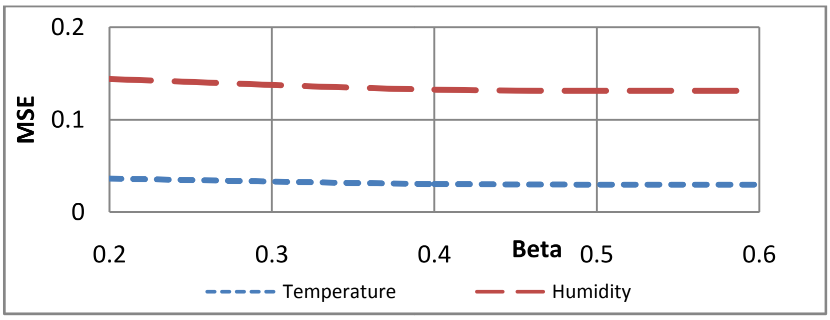
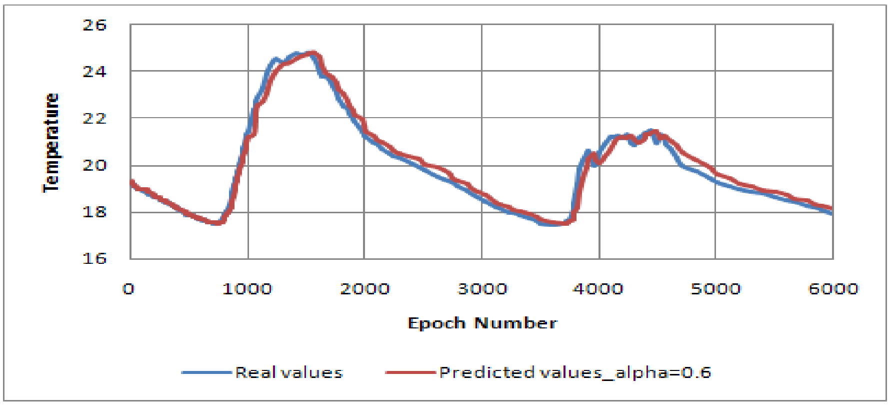
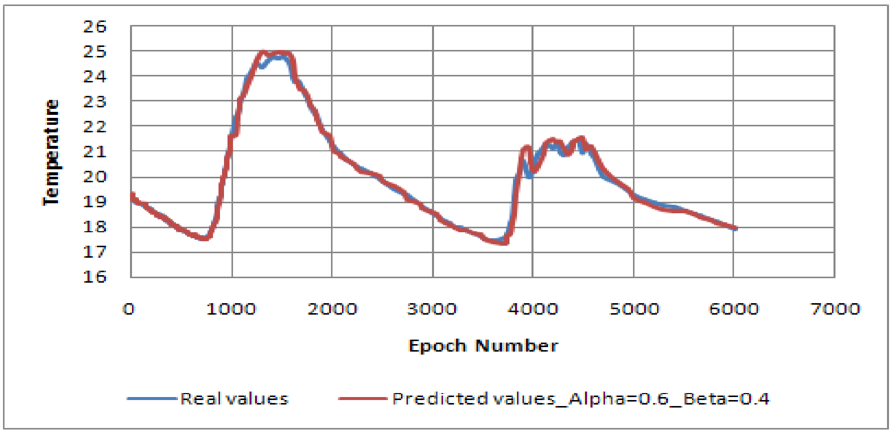


Fig. 5.6 MSE decreases as β increase.

Figs. 5.7 and 5.8 show the difference between the real measurements and the predicted values in case SES and DES for temperature and humidity readings. The results prove that the predicted data is very close if not the same as the actual data and the predictors' convergence is very fast.



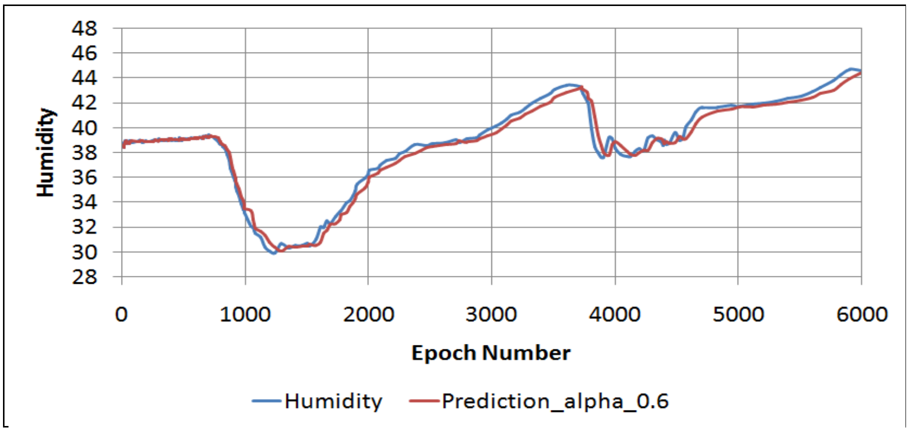
(a)



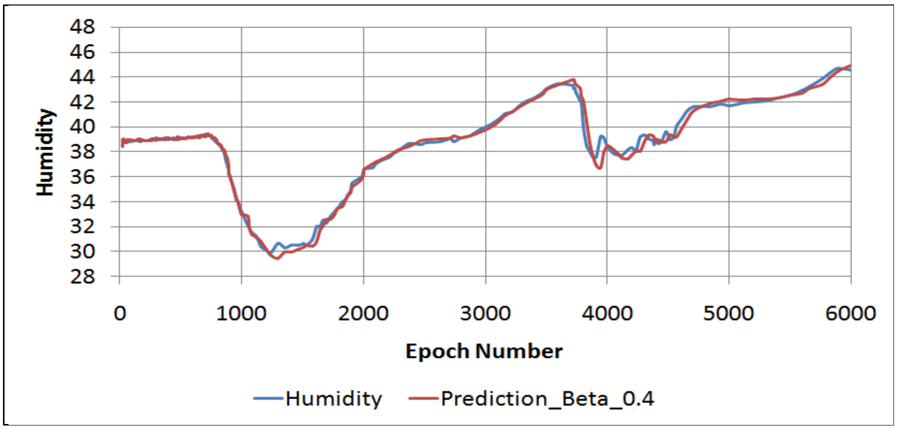
(b)

Fig. 5.7 The real temperature measurements and the predicted values in case of SES (in the top) and in case of DES (in the bottom)

However, the MSE in case of temperature values is smaller than that in case of humidity based on the used data. Therefore, exponential smoothing predictors are more accurate if the data under consideration represents temperature values rather than humidity values.



(a)



(b)

Fig. 5.8 The real humidity measurements and the predicted values in case of SES (in the top) and in case of DES (in the bottom)

5.2.3.2. The reliability test

The reliability of the measurements and the predictions is estimated based on the Standard Deviation (SD). It is sometimes evaluated as a percent of the mean, in which case it is known as a Coefficient of Variation (CV). When the measurements of a single subject are repeated to determine the mean and SD, the resulting coefficient of variation is an important measure of reliability.

CV is defined as the ratio of SD to the mean M [16]. This ratio is expressed as percent by multiplying it by 100%. CV can be employed as an indication for the data reliability. Therefore, when the slope of CV is decreasing, then the data reliability is improved [36].

$$CV = \frac{SD}{M} * 100\% \tag{5.2}$$

CV is used only for data measured on ratio scale such as humidity, time, and energy. On the other hand, it cannot be used for data on interval scale such as temperature. For instance, increasing the temperature by 1K is the same as increasing it by 1°C. As a result, SD will be the same for both scales while there will be a difference of 273 in the mean consequently, the CV will differ for the different scales.

In this work, the data reliability is investigated for humidity reading using the CV of these measurements. As shown in Fig. 5.9, CV for the humidity readings are inversely proportional to the smoothing constant (Beta) in case of DES. However, the readings are directly proportional to the smoothing constant (Alpha) in case of SES. Therefore, it is obvious that the lower the smoothing constant the higher the data reliability if the SES is used, while the higher the smoothing constant the higher the data reliability if the DES is used.

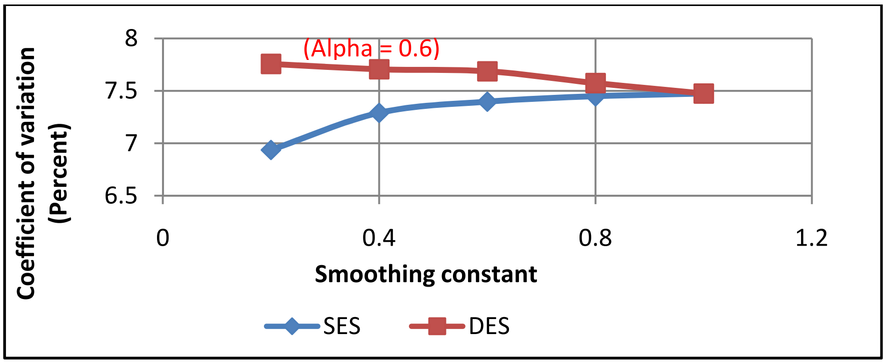


Fig. 5.9 Coefficient of variation in case of humidity readings

5.2.3.3. Simulation with random inputs

In this subsection, the data reduction proposal is test based on single and multiple smoothing predictors for both single modal and multimodal WSN. The measurements are simulated based on a random distribution as shown in Fig. 5.10. As can be seen in the figure, the fluctuation of the input values is high which might affect the operation of the predictors as given in Fig. 5.10. The lifetime of sensor networks will be investigated in three modes of operations:

1. **Normal mode:** no power saving technique is used in the WSN (naive method).
2. **Single modal:** the sensors will collect only one environmental phenomenon such as temperature or humidity. This model will be tested when SES and DES are used.
3. **Multimodal WSN:** the sensors will collect multiple phenomena such as temperature, humidity, and pressure. There are two techniques to send these measurements from the sensor node to the sink nodes. In the first technique, a sensor sends only one data message for all measurements and in the second, a sensor sends a separate message for each phenomena. Each case will be tested along with SES and DES.

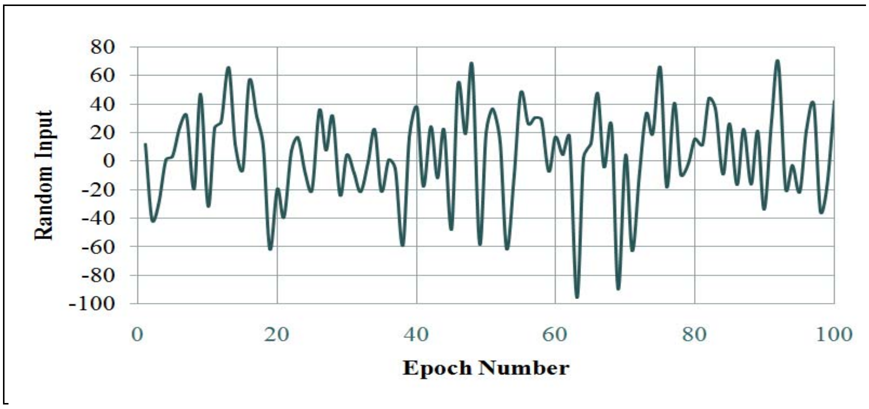


Fig. 5.10 Random input for the simulator

The results presented in this section is the average values over 1000 runs to the simulator with different settings including different network topologies. WSNs used in our simulator are considered flat where no clustering techniques are used. In addition, the logarithmic scale will be used for comparisons and the list of the lifetimes in numbers is presented in Table 5.1.

a) Case Study One: Single Modal WSN

Here, the reduction approaches is compared to the naïve communication method between the sensors and the sink node where each sensor sends its information or forward others information in a separate message. In addition, it is assumed that each sensor senses only one phenomenon from the monitored environment.

Fig. 5.11 depicts the comparison among three different algorithms, which are the naïve, SES, and DES. The three algorithms are compared in terms of their lifetimes with different threshold values. As can be seen in the Figure, DES performs almost the same as the naive model when the threshold is very low. At the same time, SES over performs the naïve model by small percentage when the threshold is low. This means the predictors' error is high relative to the threshold value. However, with increasing the threshold value, the lifetime of the network increases with large percentage since there is no need for sensors to flood the network with messages. Moreover, we noticed that the SES, in most cases, over performs the DES due to the high fluctuations in the sensors readings.

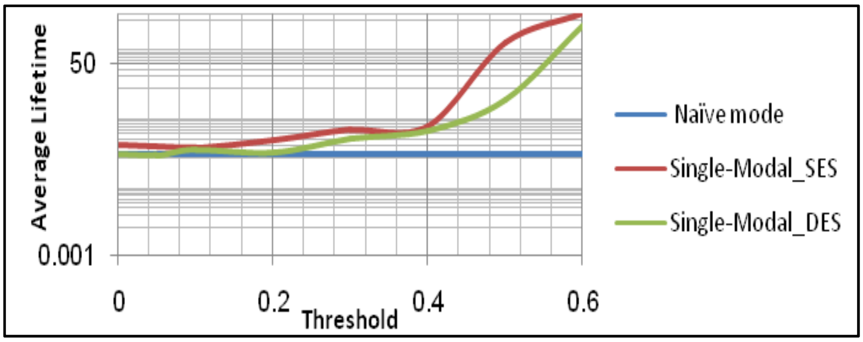


Fig. 5.11 Lifetimes in Single Modal WSN (Logarithmic scale)

b) Case Study Two: Multi-Modal WSN

In these set of experiments, the performance of our data reduction techniques are measured according to the concept of multi-modal WSNs. For the multimodal WSN, in which multiple environmental phenomena are collected. The proposed approaches are tested with two different settings, which are single packet and multi packet; in single-packet settings, the monitored data are collected and sent all together in one packet to the sink/cluster head node. While in multimodal multi-packets, each measured phenomenon is sent within a separate packet to the collector node.

Again, the SES and DES were tested with different threshold values as given in Figs. 5.12 and 5.13. For single packet settings given in Fig. 5.12, the predictors seem to perform the same as the naive model. However, after certain threshold (0.4) their performances tend to be much better than the naive model. In fact, the lifetime increases almost directly when predictors' are used.

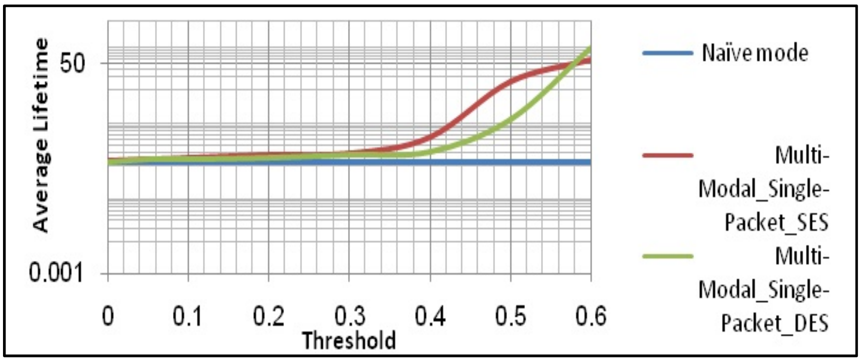


Fig. 5.12 Lifetimes in Multi-Modal WSN, Single-packet settings (Logarithmic scale)

While in the second setting, a multimodal WSN is simulated with sending the sensed features in separate messages when it is needed. Different thresholds are allowed to be used and the WSN lifetime with each threshold is recorded as shown in Figure 5.13. The performance of our approach is similar to the one presented in Fig. 5.12. In general, the previous results showed that such approaches would increase the lifetime of the network up to hundreds of times when the used threshold is approximately above 0.3.

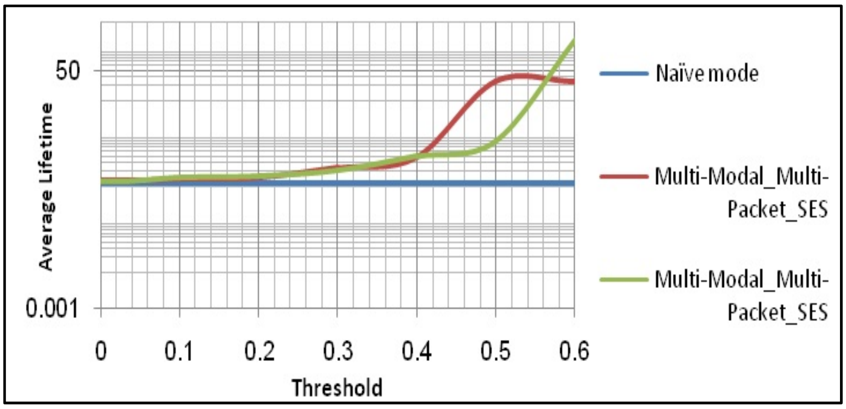


Fig. 5.13 Lifetimes in Multi-Modal WSN, Multi-packet settings (Logarithmic scale)

Table 5.1 sensor network lifetimes based on random input

			Threshold/Lifetime							
			0	**0.05**	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**	**0.6**
Naive			309							
Single Modal		**SES**	515	484.2	456.8	698.8	1248.4	1559.6	164068	902792
		DES	310.6	301.6	415	340.2	730	1211	6855.8	428508
Multi-Modal	**Single Packet**	**SES**	344.6	366.8	386.6	460.8	495.8	1118.2	19732.8	60539.6
		DES	311.4	355.2	355	386.4	446.2	515.4	2820.8	108392.5
	Multi-Packet	**SES**	341.4	335.2	366.6	379.4	611.6	996.2	30320	30138
		DES	328.6	335.8	389.2	410	530	1002.2	2034.4	206377

5.2.3.4. Simulation with real inputs

To add more reality for our experiments, the real measurements extracted from Berkley Research Lab are used as the input for the simulator. Therefore, the same simulation environment, procedure, and predictors' parameters will be used in the following experiments. In other words, the smoothing constants are set to α = 0.6, β = 0.4 according to the results obtained from previous sections. In addition, the threshold varies from 0.0 to 2, so the sensors will not send its readings as long as the error signal is smaller than the given threshold.

Additionally, the results presented in this section is the average values over 1000 runs to the simulator with different settings including different network topologies. In addition, the logarithmic scale is used for comparisons and the list of the lifetimes in numbers is presented in Table 5.2.

a) Case study one: Single modal WSN

As mentioned before, in this phase, our reduction approaches will be compared to the naïve communication method between the sensors and the sink node. Each sensor sends its information or forward others information in a separate message.

Fig. 5.14 depicts the comparison among different algorithms including the naïve, SES, and DES. The three algorithms are compared in terms of their lifetimes with different threshold values. As can be seen in the figure, the SES over performs DES for low threshold values. However, they produce almost the same performance for high threshold values. At the same time, both of them over perform the naïve modal for all threshold values. This means that predictors' error is small relative to the threshold value. In addition, with increasing the threshold value, the lifetime of the network increases with large percentage since there is no need for sensors to flood the network with messages. However, the reliability of the data will be degraded.

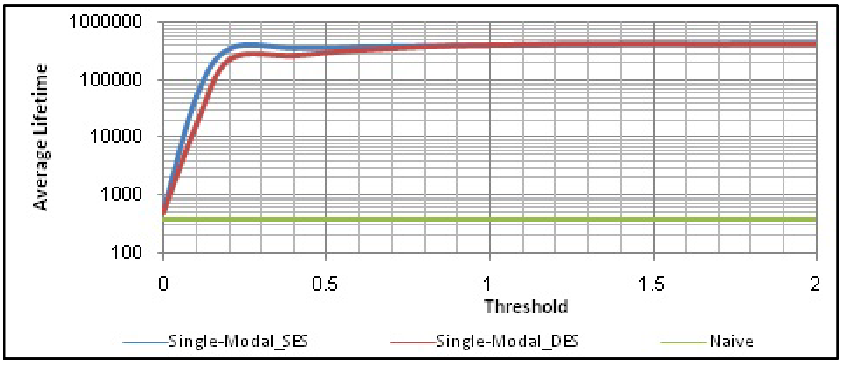


Fig. 5.14 Lifetimes in Single Modal WSN (Logarithmic scale)

b) Case study two: Multi-modal WSN

This approach is tested with two different settings, which are single packet and multi packet. For single packet settings, SES seems to perform the same as DES as shown in Fig. 5.15. At the same time, their performances tend to be much better than the naive model. In fact, the lifetime increases almost linearly when predictors' are used until threshold reaches 0.3 then it becomes constant beyond this threshold.

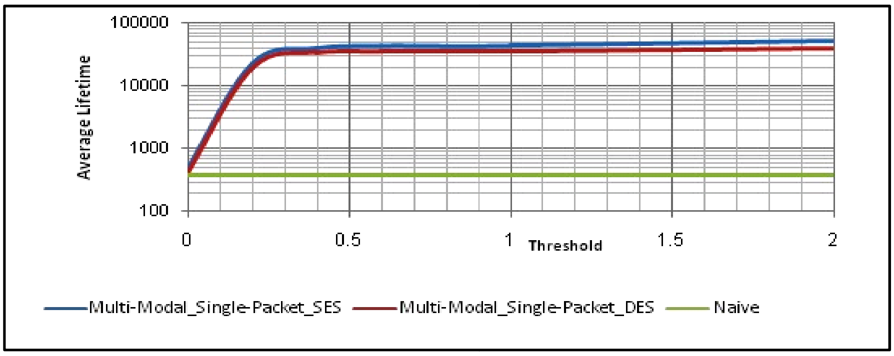


Fig.5.15 Lifetimes in Multi-Modal WSN using real sensor readings (Single-packet settings)

As shown in Fig. 5.16, in the second setting, a multimodal WSN is simulated with sending the sensed features in separate messages when it is needed. The performance of our approach is similar to the one presented in Fig. 5.15 except the fact that SES over performs DES for threshold values below 0.4.

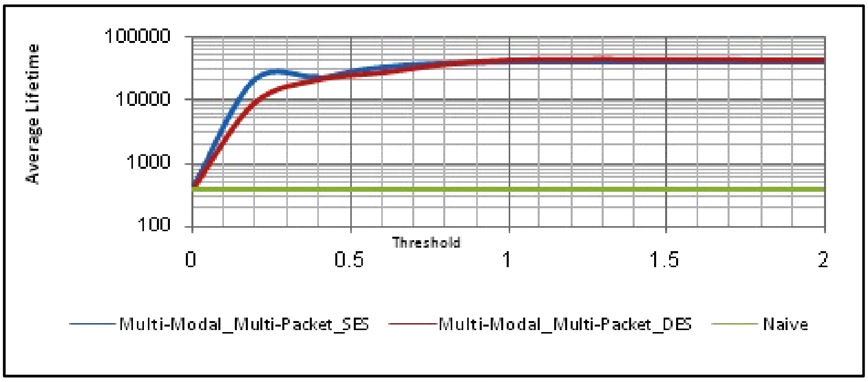


Fig. 5.16 Lifetimes in Multi-Modal WSN using real sensor readings (Multi-packet

Table 5.2 Lifetimes with real input

			0	0.2	0.4	0.6	1	2
Naive			391.4					
Single-Modal		**SES**	554	356146	365395	377551	409453	428667
		DES	525	225671	264225	336176	414774	422734
Multi-Modal	**Single Packet**	**SES**	504	23440	39792	43208	43262	52327
		DES	433	19757	34490	36199	36292	38878
	Multi-Packet	**SES**	396	20086	22720	33665	38390	39711
		DES	392	8870	20272	27079	41206	41891

5.2.4. Data Reduction Based on Fuzzy Logic Algorithm

Fuzzy logic is used as an alternative solution to the problem of power consumption in WSNs through minimizing the number of messages sent to the end user. In this case, sensor nodes turn off their transceivers until a real danger is probably exist. This danger is sensed based on temperature, humidity, and light intensity measurements.

The first step in the simulation process is to define the linguistic variables and terms. So, for the environmental parameters “Low”, “Medium” and “High” terms are used as the linguistic input variables, while “Very Low”, “Low”, “Medium”, “High” and “Very High” terms are used for the linguistic output variable (probability of fire). Afterwards, the membership function is constructed as shown in Figs. 5.17 and 5.18 where the membership function for the humidity and the light intensity is the same as that for the temperature.

Third, the rule base is constructed while the crisp input data are converted to fuzzy values using the membership functions. Then, the rules are estimated in the rule base. Finally, the results of each rule are combined and the output data are converted to non-fuzzy values using the center of gravity for singletons method. Table 5.3 shows sample of the fuzzy rules used in the defuzzification process to determine the probability of fire.

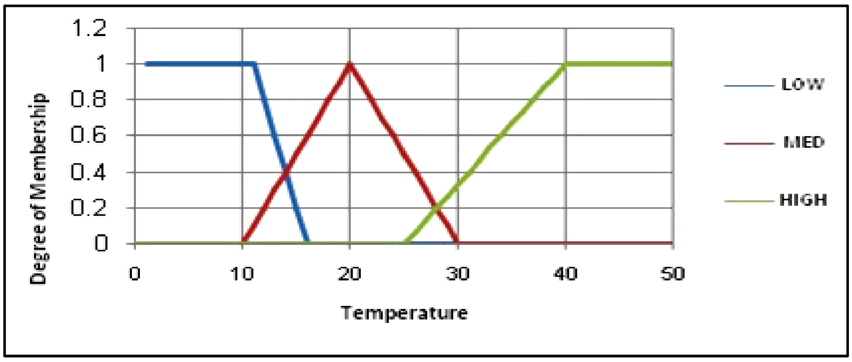


Fig. 5.17 The membership function for the input variable

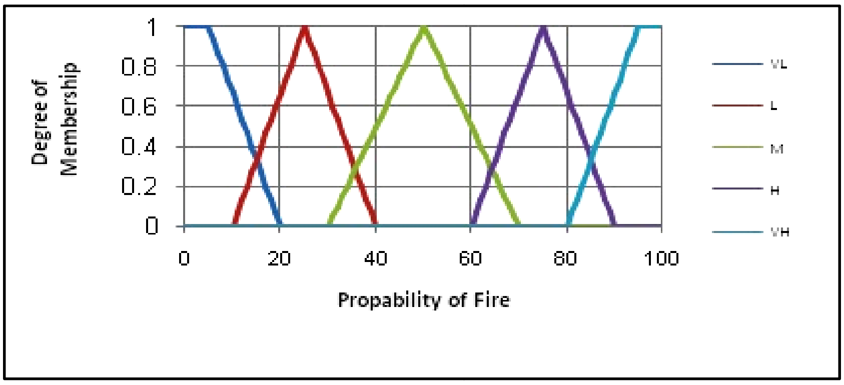


Fig. 5.18 The membership function for the output variable

The simulator parameters are the same as those in the previous section, and the real data extracted from Intel Berkley Research Lab are used as the measurements of the sensor nodes. As shown in Fig. 5.19, the average lifetime of the WSN increases as the probability of fire increase except for the region between 55% and 59%, the lifetime is approximately constant. Through comparing Figs. 5.14, 5.15, and 5.16 with Fig. 5.19, it is obvious that, the average lifetime of the WSN based on fuzzy logic system at 60% probability of fire, for instance, is fairly double that of the single modal WSN and is approximately 17 times that of the multi modal WSN based on exponential smoothing predictors.

Table 5.3 Examples of fuzzy Rules

Temperature	Humidity	Light Intensity	Output
L	H	L	VL
L	M	H	L
H	M	M	H
H	L	H	VH
:	:	:	:

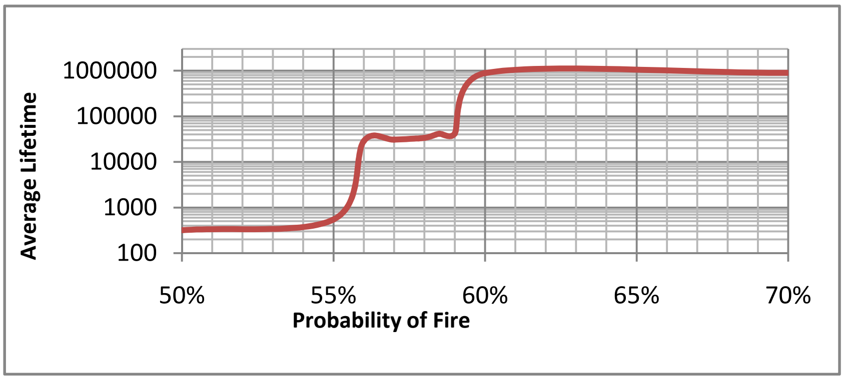


Fig. 5.19 The average lifetime of WSN for different probability of fire

The result presented in [60] is based on the implementation of LMS adaptive filters as predictors with threshold 0.5°C. Fig 5.20 shows a comparison among our approaches and the approach presented in [60]. The values in Fig. 5.20 represent the percent of increase of the overall network lifetime. The figure shows that fuzzy logic approach provides larger saving in power consumption compared to other approaches.

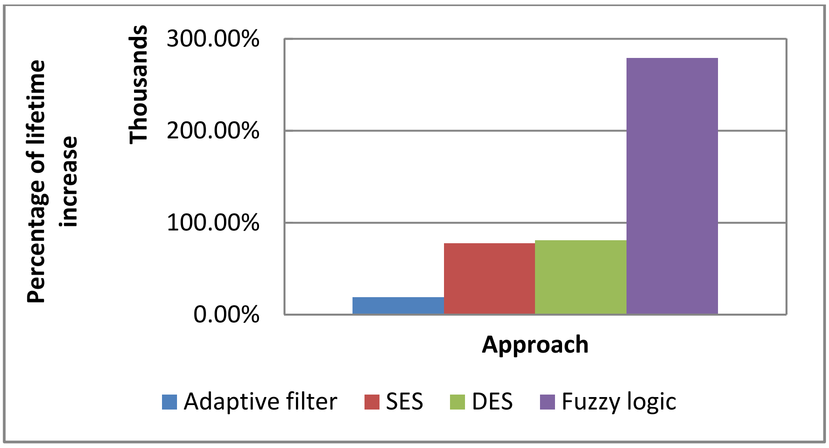


Fig. 5.20 comparison of results from different approaches

5.2.5. Threshold and Tolerance Based approaches

In this section, the data reduction techniques are implemented based on simpler ways. One example of this is to use a certain value as a threshold below which the nodes will turn off its transceiver. However, if this threshold is exceeded, the nodes send a message to the base station. The simulator parameters are the same as those in the previous section, and the real data extracted from Intel Berkley Research Lab are used as the measurements of the sensor nodes.

As shown in Fig. 5.21, the lifetime increases directly when the temperature readings exceed 20 degree. Moreover, the lifetime of WSN based on this approach is approximately half that of the WSN based on fuzzy logic. The advantage of this technique is its ability to prolong the WSN lifetime. On the other hand, it cannot provide information about the sensed phenomenon continuously.

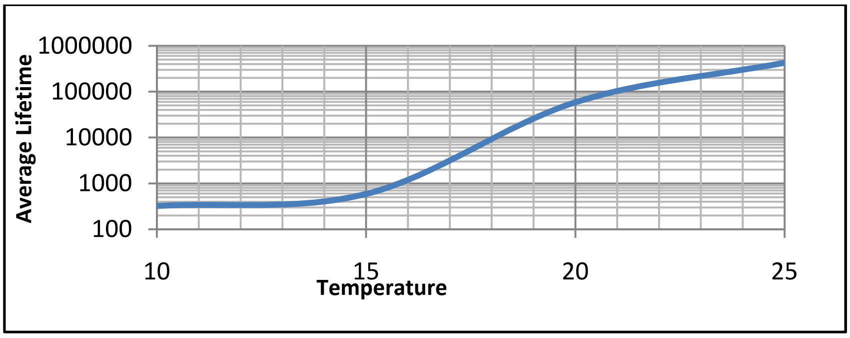


Fig. 5.21 Lifetimes in Single Modal WSN based on a given threshold

Another technique employs the difference between the present and past readings in order to reduce the radio messages. In this case, the nodes undergo in the sleep mode as long as this difference is lower than a certain tolerance defined by the user. As shown in Fig. 5.22, the average lifetime increases gradually when the tolerance exceeds 0.2. Then, the rate at which the lifetime increase becomes slow when tolerance exceeds 0.4. Again, the fuzzy logic algorithm over performs this approach since the lifetime of the WSN based on this approach is fairly half that of the lifetime of WSN based fuzzy logic algorithm.

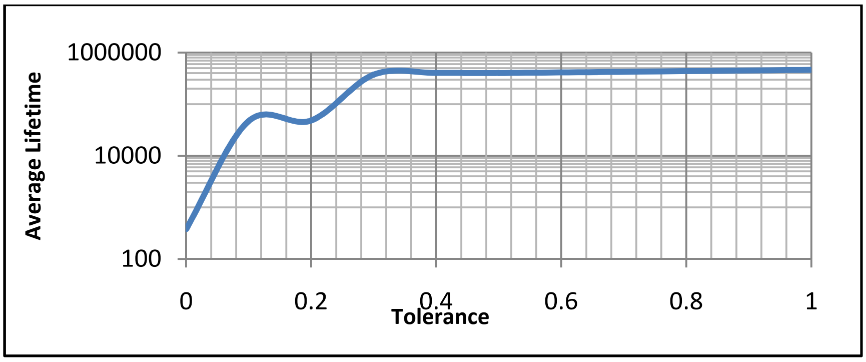


Fig. 5.22 Lifetimes in Single Modal WSN based on the changes in the readings

5.3. Experimental Results

In this section, the exponential smoothing predictors on a real WSN are tested. National Instruments (NI) WSN kit is used for this purpose [62], where its sensor nodes are programmable in order to download the proposed algorithm on its flash memory and then watch its behavior for a period. As shown in Fig. 5.23, the kit contains an NI WSN-9791 Ethernet gateway, two battery powered programmable nodes (NI WSN-3202 ±10 V analog input node, and NI WSN-3212 thermocouple input node).

The operations inside each node are controlled by a TI MSP430 microcontroller that is designed for low-power embedded systems. One of the advantages of NI WSN kit is the ability to send and receive string-based user messages to and from the node to debug the deployed application. In addition, it allows the user to optimize node behavior in terms of sample rate versus battery life. The gateway works as a bridge between the IEEE 802.15.4 wireless network and the wired Ethernet network. In addition, it is responsible for node authentication, and message buffering. The default behavior of an NI node is to sample all channels and transmit every sample acquired to the gateway.

Fig. 5.23 NI WSN architecture [61]

However, intelligence is added for the nodes to increase their lifetime by extending the transmit interval which is equal to the period between the successive transmissions of a certain sensor node. Each sensor node contains TI MSP430 microcontroller, which controls all the functions, 4 AA cells Battery, flash memory, and radio transceiver operates at frequency 2.4 GHz. The radius of the sensing field that can be covered by the node is extended to 300 m using IEEE 802.15.4 wireless networking protocol especially it is suitable for low data rate networks. In addition, it enables the nodes to switch to the sleep mode to conserve power and still maintain reliable communication.

The measurement nodes are programmed using NI LabVIEW, which is graphical programming environment used to develop measurement, test, and control systems using graphical codes. These codes have become the industry standard for developing applications that acquire and process data. LabVIEW software can be integrated with many hardware devices. Therefore, it can be used to develop real time applications with the aid of different toolkits and drivers.

The experiments are divided into two stages. In the first one, the performance of SES and DES predictors is tested on LabVIEW using damped sinusoid input to ensure that the error between the real and the predicted values is small. Fig. 5.24 depicts the Virtual Instrument (VI) block diagram. The VI block diagrams include data processing modules, and predefined mathematical models such as adders, substructures, integrators, etc. used in a mixed modality platform. In the block diagram, there is a terminal for every object created in the front panel. A signal generator is used to produce a damped sinusoid input, which triggers a formula node. This node contains the predictors' equations that produce the measurement at time t+1. Afterwards, the error signal is computed and compared with the given threshold (0.5 degree). In these simulations, the smoothing constant (alpha) was adaptive rather than using fixed value [80]. Actually, they are programmed to adapt to the resultant error according to (5.3), (5.4), and (5.5). However, this method did not support the function of the algorithm as a real time predictor since it forced alpha to approach one to eliminate the error consequently we lose the ability of the predictor to smooth out random fluctuations.

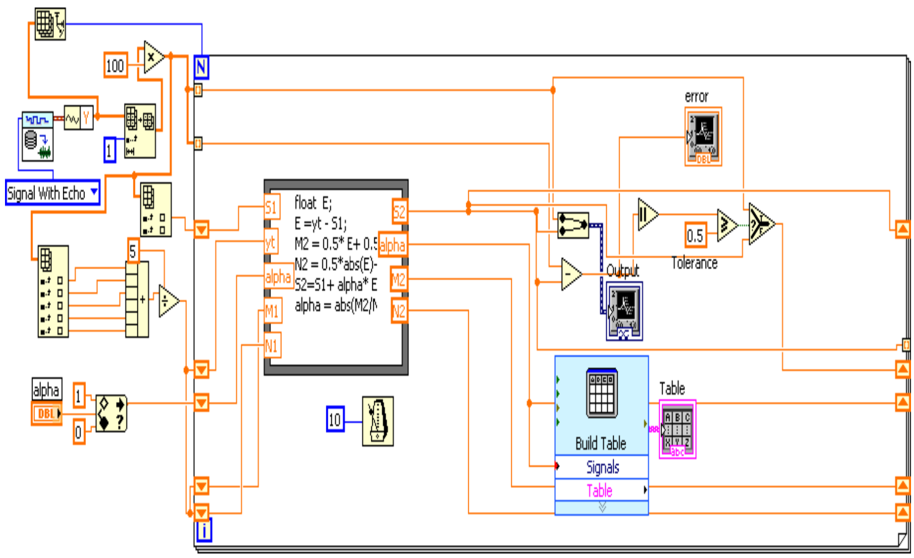


Fig. 5.24 Block diagram of the implementation of SES in LabVIEW (VI)

$$E_t = r\, e_t + (1-r)\, E_{t-1} \tag{5.3}$$

$$A_t = r\left|e_t\right| + (1-r)\, A_{t-1} \tag{5.4}$$

$$\alpha = \left|\frac{E_t}{A_t}\right| \tag{5.5}$$

Fig. 5.25 shows the front panel of the predictors where it compares between the predicted and the actual values, and the error signals are drawn for these two predictors. It is obvious that for both of them that the error is large in the learning phase. However, it approximately falls to zero after small number of samples. In fact, the behavior of the predictors is the same except that DES provides more smoothed signal than that produced by SES.

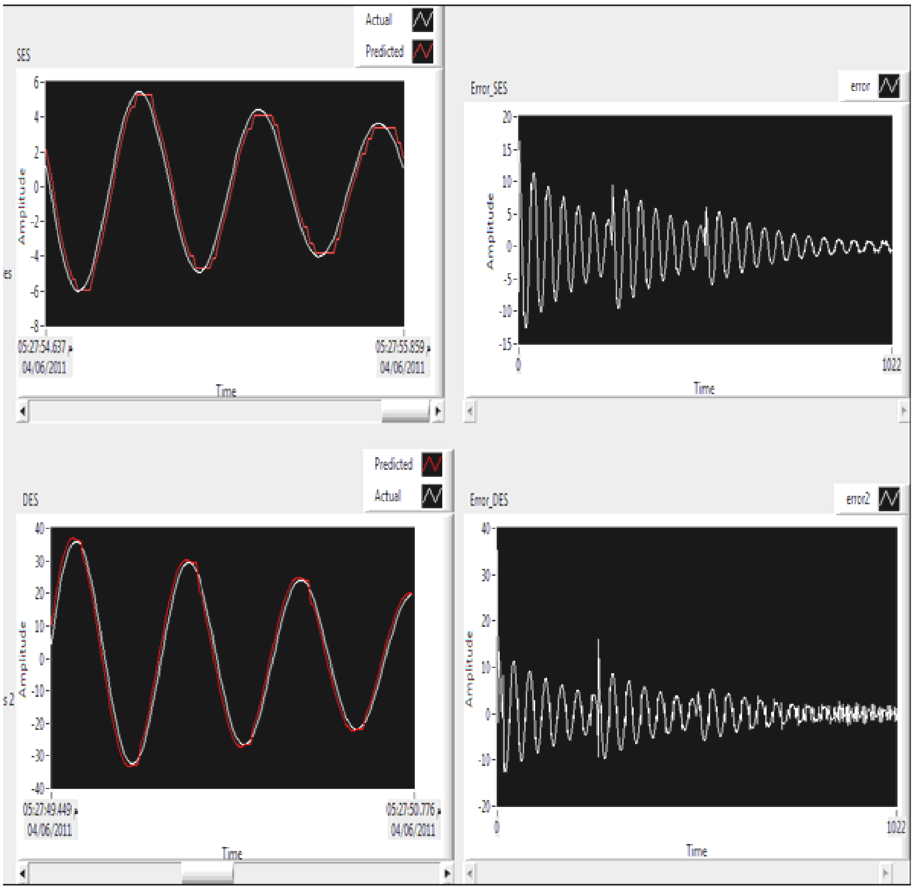


Fig. 5.25 Front panel for the SES and DES predictors

Finally, the graphical code for the DES predictor is implemented as shown in Fig. 5.26. Subsequently, we downloaded it to the sensor nodes but at the same time, an identical predictor is running on the host controller. The lifetime of the node is determined based on the sample interval, and the transmit interval.

In our experiment, the sampling rate is set to constant value, for instance 5Hz, while the transmit interval is adaptive where this period is determined based on the error between the actual measurements and the predicted values. For example, if the error signal is below 0.5 degree, the transmit interval is set to 5000 second. While if the error signal is below 1 degree, the transmit interval is reduced to 50 second. However, if the error increased beyond 1 degree, the nodes transmit every 1 second. However, from our observations, the nodes transmit the measurement every 5000 second due to the ability of the predictor to follow the real values especially in case of temperature measurements whose values change slowly.

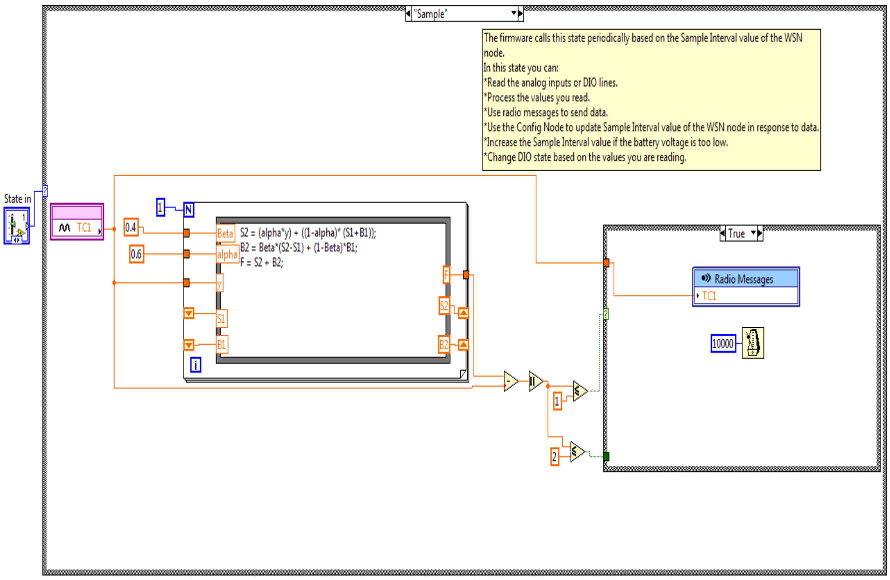


Fig. 5.26 Controlling the transmit time by the DES predictor (VI)

From the LabVIEW WSN pioneer performance benchmark, it is stated that the lifetime of the nodes is 5.3 month when the transmit interval is set to 5 seconds. On the other hand, our approach increases the lifetime to 27 month for transmission interval of 5000 second. However, longer lifetime can be obtained if the transmit interval is extend beyond 5000 second.

5.4. Discussion of the Results

The obtained results in this work show that the overall lifetime of WSNs can be extended through the dual prediction scheme. DPS can be implemented using different types of predictors. However, Exponential Smoothing predictors are used in this work and their performance is tested before applying them into the DPS.

The results of this test show that these predictors are good candidates for the DPS since it achieves high accuracy with small memory footprint and very low computational overhead for instance only one past value is required to produce the prediction. In addition, these predictors take into consideration the different components of the data including trend and seasonality components. On the other hand, data reliability depends on the level of prediction and on the smoothing constant. It is found that the reliability has an increasing slope when DES is used while the slope has decreasing when SES is used.

Simulations of WSN with and without the DPS are made twice. Firstly, the measurements are simulated using random generated values to test the ability of the predictors in following the high fluctuations. Fortunately, the predictors succeeded in prolonging the lifetime hundreds times. Secondly, it was important to examine the predictors using real data. Again, the results were consistent and showed that both of SES and DES approximately provide the same performance. As a result, it is recommended to use DES rather than SES since it achieves two important criteria including long lifetime and improved data reliability.

Additionally, data reduction can be implemented using Fuzzy Logic algorithms. It is found that this method achieves the highest increase in the overall lifetime. On the other hand, it is limited only to multi-modal networks. Two other simple techniques are prototyped for comparison purpose and they achieve lower lifetime than that obtained using the fuzzy logic algorithm. Additionally, a comparison among the proposed techniques and another one based on LMS adaptive filters is made. It is found that the proposed approaches in this work over perform the LMS algorithm approach.

Chapter 6

Real Life Application: Greenhouse

6.1. Introduction

Greenhouses are used by botanists, commercial plant growers, and dedicated gardeners. Particularly in cool climates, greenhouses have the advantage of allowing sunlight to enter and preventing heat from escaping. Consequently, they are useful for growing and propagating plants since they provide the plants with the suitable environmental conditions such as temperature, humidity. This can be achieved through the transparent covering, which allows visible light to enter the greenhouse. Afterwards, the heat is absorbed by the materials of the covering and is prevented from leaving by reflecting the energy back into the interior. In addition, greenhouses may be used in the winter to increase the temperature by making some modifications on the covering of the greenhouse.

In this chapter, a number of applications for the WSN in the field of agriculture are proposed. However, our main contribution in the field of agriculture is the implementation of a monitoring system inside greenhouses. This system has the ability to monitor the environmental parameters inside the greenhouse such as temperature, humidity, soil moisture, solar radiation, and nitrogen level. Afterwards, it decides which disease may infect the plants due to these parameters. Two architectures of the monitoring system are proposed, called GreenSense-1 and GreenSense-2. They differ in the nature of the data sent from the sensor node to the sink node outside the greenhouse. However, both of them utilize the fuzzy logic controllers that process the readings come from the sensor nodes. These controllers are used due to their simplicity and flexibility where they can handle problems with incomplete data. In addition, fuzzy controllers are employed to match any set of input-output data if the system is changing. The Fuzzy Logic Toolbox in Matlab will be utilized for the implementation by supplying adaptive techniques such as adaptive Neuro-fuzzy inference systems (ANFIS).

6.2. Related Work

The problem of controlling and monitoring the environmental parameters inside greenhouses is addressed in many researches. For instance, M. El Aoud, et al proposed in [57] implemented an intelligent climate controller based on the fuzzy logic. In their work, a model was proposed to simulate the internal state of greenhouse in order to test algorithms of control. In addition, to make the system adaptive with the changes in the surrounding environment, they proposed an optimized fuzzy controller based on the gradient descent to modify the controller's parameters automatically as shown in Fig. 6.1.

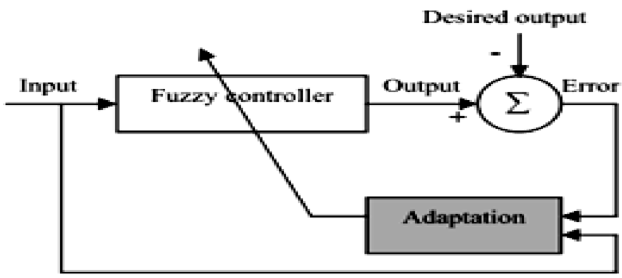


Fig. 6.1 Adaptive fuzzy logic controller

G. Saridakis et al proposed in [30] an indoor environment and energy management system for microclimate control of greenhouses. The monitored variables are the greenhouse's indoor luminance, indoor temperature, relative humidity, CO_2 concentration, and the outside temperature. While the actuators are the heating units, motor-controlled windows, motor controlled shading curtains, artificial lighting, CO_2 enrichment bottles, and water fogging valves. The platform used is the Lon-Works platform. The controllers were implemented in Matlab and were interconnected into the system as a separate node by taking the advantages of the Lon-Works protocol. The fuzzy controller is separated in two controllers for design and presentation simplicity. FLC-1 aims to control the CO_2 concentration and the radiation inside any greenhouse using actuators such as CO_2 enrichment bottles, shading curtains and artificial lighting. On the other hand, FLC-2 tries to obtain the desired temperature and humidity in the interior of the greenhouse. Moreover, they installed an i-Lon server in the Lon-Works network. This Internet server offers monitoring and control of all devices where it can access devices from a local network, a virtual private network, or the Internet in order to manage, monitor, and collect the necessary data.

P. Lanfang et al proposed in [69] a technique for obtaining the nonlinear functional relation between the greenhouse temperature and various climate factors through combining physical model of the greenhouse climate with adaptive fuzzy logic system. They mentioned a model for greenhouse climate however many parameters are hard to determine. The output of the proposed fuzzy algorithm depends on the outside temperature, wind speed, solar radiation, and the relative humidity of greenhouse. In addition, the back propagation learning algorithm is used to train the fuzzy controller by determining the error between the fuzzy output and the output of the physical model.

In addition, M. Panoiu et al proposed in [58] a technique for maintaining optimal environmental conditions for a greenhouse for seedlings of fir trees based on fuzzy logic and they used a DSP development board. They stated the optimal climate conditions for fir trees as follow: soil moisture (50%: 80%), temperature (10°C: 35°C), and humidity (70%: 90%). The inputs of the fuzzy algorithm are the ambient temperature, air humidity, and soil moisture while the outputs are the soil wetting, heating witch, and steam spray. Moreover, they simulated the fuzzy controller in simulation software XFUZZY (Java) as well as performed real experiments using DSP board.

P. Javadi, et al proposed in [68] an irrigation controller that manages the flow of water and fertilizer to the plants. The controller is based on the fuzzy logic model and it is built on MATLAB software. They utilized the concept of closed-loop control where the controller receives feedback from the sensors in the field. These sensors provide updated data to the controller about parameters that are affected by the system behavior (such as soil moisture level, temperature in hothouses and so on). They used inputs such as soil moisture, temperature, humidity, wind speed, radiation, and salinity while the outputs would drive the valves, energy systems (lights, heating, and ventilation), open or close walls, and roofs of the greenhouse.

In the simulations, they modeled the temperature, humidity, radiation as sine wave with frequency of 0.2618 rad/h. This frequency is measured according to a period of 24 h. The authors used the FAO Penman-Monteith method to determine the desired soil moisture. Moreover, the authors made a comparison between the on/off controller and the fuzzy one. The result showed that the measured soil moisture follows the desired one without any oscillations in case of fuzzy controller is used. In addition, the output of controller in case of on/off controller is used tracks the desired one but with high oscillations, which in turn increases the increase the energy consumption.

I. Hameed proposed in [39] simple architecture for type-2 fuzzy logic controller to handle uncertainties where type-1 FLC is not able to handle these uncertainties. He showed that it is impossible for a model to capture all the characteristics of the actual plant. Therefore, the performance of a controller designed using a model will be reduced when it is applied to the actual plant. Consequently, he used T2FLC to handle modeling uncertainties. However, T2FLC suffers from the high computational overhead. Hence, Hameed proposed a simple way to reduce the computational burden by decomposing the T2FLC into four T1FLC and implemented the model in MATLAB. Genetic algorithm (GA) is used to optimize the controller parameters and is used as an uncertainty sensor to detect the level of uncertainty. Additionally, the author proposed a simple greenhouse Heating–Cooling Ventilating (HCV) model for controlling temperature and humidity ratio inside a greenhouse by means of heating, ventilating, and humidifying the air within it. To achieve that, a fuzzy pseudo-derivative feedback controller is used.

F. Lafont et al presented in [25] three techniques to determine automatically the rules aid of input/output measurements (Takagi-Sugeno type fuzzy models, optimization of Fuzzy Regression Trees, Iterative fuzzy modeling). They used a dynamic physical knowledge model to simulate the behavior of the greenhouse and they used an identification model of greenhouse based on square-means. They presented a model based on fuzzy logic with four inputs and three outputs. These will require 81 rules but if the inputs increased, the fuzzy controller becomes unmanageable. Therefore, they developed a decentralized control structure including two robust fuzzy controllers to reduce the rules. The first one is designed for the heating and the roofing with the temperature input while the second one for the moistening with the hygrometry input. Thereby, they succeeded in reducing the rules 16 rules only.

6.3. WSN Applications in Agriculture

In this work, the research is limit to the ornamental diseases that may result due to unsuitable environment conditions. However, the concept can be applied to other plants that grow inside greenhouses. Table 6.1 summarizes some of these diseases and the corresponding temperature and relative humidity that develop these diseases [65]. All diseases have a specific range of temperatures and relative humidity percentages under which they are the worst. In addition, other parameters such as soil moisture, nitrogen level, and shadowing significantly influence the infection of the plants. Therefore, WSNs are able to detect these harmful conditions and report a decision to the sink node outside the greenhouse to automatically trigger a controller, which in turn orders the mist or other devices to operate to keep the environmental conditions outside these harmful ranges.

Table 6.1
Sample of ornamental diseases [65]

Disease	**Temperature**	**Humidity**	**Soil moisture**	**Nitrogen**	**Shadowing**
Gray leaf Spot	70°F : to 95°F	Greater than 70%	Low	High	Semi-shadow
Powdery Mildew	68°F : 77°F	40%: 70% (Daytime) 95% : 99% (Night)	Low	High	Shadow
Botrytis leaf blight	70°F : 77°F	Greater than 85%	High	High	Shadow
Downy Mildew	60°F : 74°F	Greater than 85%	High	Low	Shadow
Rust	59°F : 77°F	Greater than [illegible]	Low	Low	Sunlight

Moreover, some applications in the field of agriculture are proposed. For instance, Bermuda grass is a plant that is used in Stadiums, clubs, and tourist villages. This plant is sensitive to temperature degrees where it is subjected to yellowing when the surrounding environment has small temperature degree. Consequently, WSN can be used to indicate the changes in temperature in order to protect this grass from yellowing. For example, stadiums located near the seashore are subjected to changes in temperature degrees all day. Here, WSN may be used to help workers to make a certain decision about the grass in order to protect it from the great changes in the temperature degrees.

Generally, the types of plants can be divided into two categories. One of them is the plants that grow in the winter while the second type grows in the summer. Therefore, there is a need for storage system that enables us to have these products all the year. However, some of the plants need small temperature during storage phase such as apple where its optimum storage temperature degree is ranged from 0 to 3°C. Other plants require relatively high temperature degrees during storage phase such as banana and Guava where their optimum storage temperature degree is ranged from 13 to 15°C [64]. WSN can be used to ensure the actual temperature inside the storage areas to allow the user to make a decision if the temperature degrees decreased or increased beyond the desired ranges. In addition, the storage place sometimes has a large area, which allows the contrast of the temperature degrees inside this place. Therefore, many sensor nodes can be used inside this area to detect the points at which there is a probability of danger.

Another application tries to provide the suitable environment for the plants inside the greenhouse during the germination and propagation. The plants inside greenhouse can be propagated by many ways. One of these ways includes exposing the roots to a specific temperature degree. For better propagation, one wants to maintain good humidity, adequate warmth, and less than full sun. This process will be adequate if the environmental parameters are monitored continuously by means of WSN.

In the last proposed application, the percentage of Oxygen and Carbon Dioxide inside the greenhouse will be continuously monitored. Carbon Dioxide (CO_2) plays an important role during photosynthesis where it is combined with water with the aid of light energy to form sugar. Some of these sugars are used to obtain complex compounds that increase dry solid plant substances for continued growth to final maturity. However, when the supply of CO_2 is reduced below certain threshold, the complex plant cell structure cannot utilize the sun's energy fully and growth or development is curtailed. The amount of CO_2 a plant requires to grow may vary from plant to plant; for instance, most plants will stop growing when the CO_2 level decreases below 150 ppm. Even at 220 ppm, a slow-down in plant growth is significantly noticeable [15].

As shown in Fig. 6.2, sensor nodes can be utilized to continuously measure the percentage of CO_2 and O_2 then send these measurements wirelessly to the farmers to allow them make the suitable decision in case of the reduction of gases. This is essential for the workers since the temperature degree inside the greenhouses often exceed 50°C. therefore, it will be hard for them to stay there for long period to measure these quantities especially when the area of the greenhouse is large.

Fig. 6.2. Deploying sensor nodes inside greenhouse (ornamental plants)

6.4. GreenSense Solution

In this section, our WSN architecture for Greenhouse monitoring is described. As mentioned in chapter 5, the NI sensor nodes are connected wirelessly to a gateway inside the Greenhouse. The Gateway is connected through a wired Ethernet to a server that is connected to the Internet with static IP address. A simple monitoring tool is built to detect and report the diseases that may infect the plants inside the Greenhouse. Based on our REDR energy saving technique, although sensor nodes do not send their data frequently, the tool is seamless to the user.

The approach is used to develop a prototype for a system that is able to monitor the important parameters inside the greenhouse including temperature, humidity, soil moisture, light intensity, and level of nitrogen. However, two schemes for the monitoring system are proposed.

In the first one, called GreenSense-1, sensor nodes are deployed inside the greenhouse where the function of these nodes is to report the environmental parameters without any internal processing except the REDR algorithm that runs for maximizing the lifetime of the deployed sensor nodes. Simultaneously, the sink node is connected to a host controller such as PC or DSP board. This controller process these measurements using fuzzy logic algorithm to determine the probability of infection for the diseases mentioned in Table 6.1. Afterwards, the framers will make a decision based on the resultant probabilities. FLC is chosen because it consist several advantages compared to the other classical controller such as simplicity of control, low cost and the possibility to design without knowing the exact mathematical model of the process.

In the second approach, called GreenSense-2, an additional mission is added for the deployed sensor nodes by implementing a fuzzy logic algorithm in these nodes to either determine the probability of infection for each disease as shown in Fig. 6.3 or generate the degree of infection for the plants inside the greenhouse. However, the nodes will not report any information to the sink node as long as the output of the fuzzy logic algorithm lies below a certain safe threshold. Consequently the nodes' lifetime maximization is achieved through a threshold based approach rather than employing the REDR algorithm.

6.5. Experimental Results

In this section, the fuzzy logic algorithm for detecting and reporting the infection probabilities of the mentioned diseases will be presented. As shown in Fig. 6.3, the FLC is decomposed into five controllers. Each one is responsible for detecting the environmental conditions that helps in developing a certain disease. Therefore, the following steps will be repeated for each controller according to the disease and its developing environment.

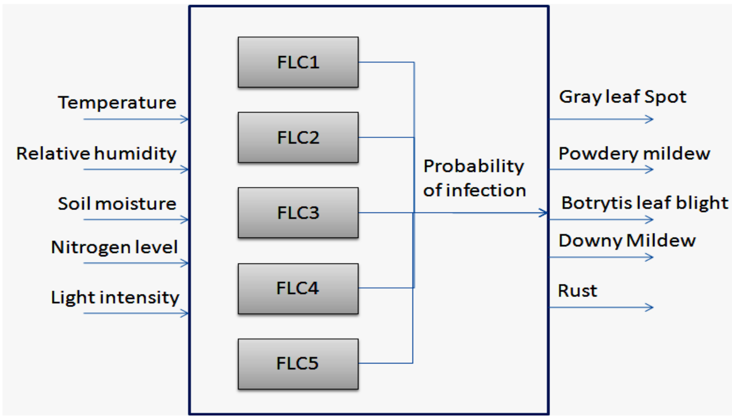


Fig. 6.3 The fuzzy logic controller

6.5.1. GreenSense-1

In this proposed scheme, each FLC is designed as multi-input, single output (MISO) system where the inputs are the environmental conditions with only one output that represents the probability of infection. Mamdani reasoning technique will be used for the inference system since it is more intuitive, has widespread acceptance, and is better suited for human input.

The first step is to define the linguistic variables and terms. For instance, for the temperature "Cool", "Warm", and "Hot terms are used as the linguistic input variables and "Low", "Medium", "High" are used for the humidity, the soil moisture, the level of nitrogen while terms such as "Shadow", "Semi-shadow", and "Sunlight" are used for the light intensity. On the other hand, "Low", "Medium", and "High" terms are used for the linguistic output variable (probability of infection). Afterwards, the membership functions are constructed as shown in Figs. 6.4 and 6.5.

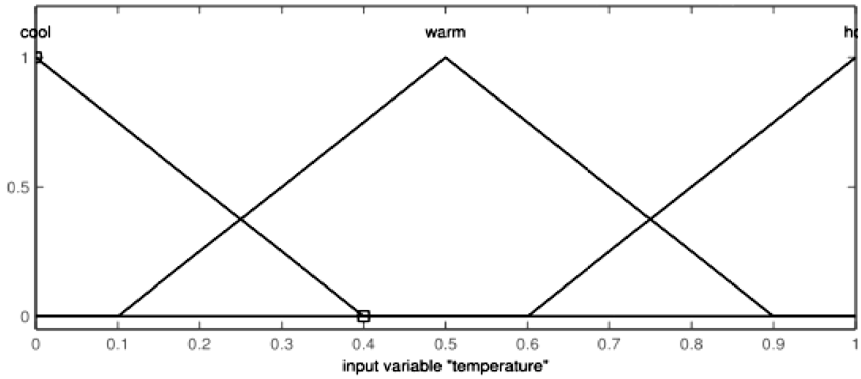


Fig. 6.4 Labels and membership functions of the temperature readings

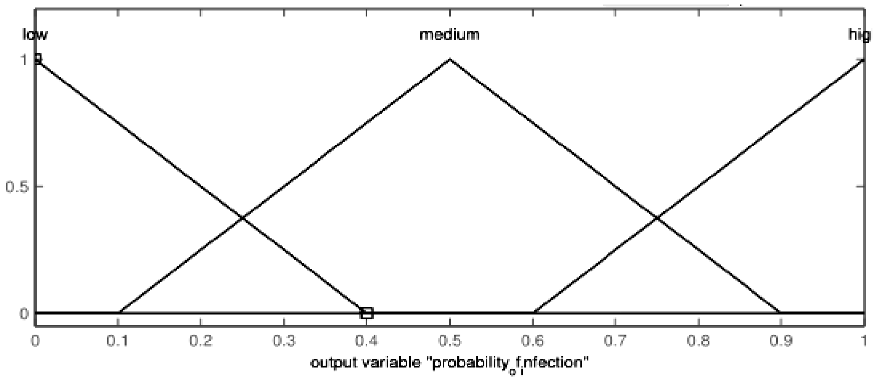


Fig. 6.5 Labels and membership functions of the output variable

Third, the rule base is constructed and the crisp input data are converted to fuzzy values using the membership functions. Then, the rules are estimated in the rule base. Table 6.2 shows sample of the fuzzy rules used in the defuzzification process to determine the probability of disease occurrence. After evaluating the result of each rule, these results should be aggregated to obtain the system's output. The maximum algorithm is generally used for accumulation as in equation (6.1).

$$Max\{\mu_A(x), \mu_B(x)\} \tag{6.1}$$

With μ_A and μ_B are the membership functions for fuzzy sets A and B. Finally, the output data are converted to non-fuzzy values. Defuzzification is performed according to the membership function of the output variable. The center of gravity for singletons algorithm given in (6.2) is used in the defuzzification to produce the output value.

$$U = \frac{\sum_{i=1}^{p}[u_i \mu_i]}{\sum_{i=1}^{p} \mu_i} \tag{6.2}$$

Where U is the result of defuzzification, u is the output variable, p is the number of singletons, μ is membership function after accumulation, and i is the index.

Afterwards, our proposed algorithm is tested on National Instruments (NI) WSN kit [62]. As mentioned earlier, the kit contains two battery powered programmable nodes, an Ethernet gateway, thermocouples, humidity sensors, and gas sensors.

Table 6.2 Sample Fuzzy Rules

Temperature	Humidity	Soil moisture	Nitrogen	Light intensity	Probability of infection
Cool	Low	high	Medium	Semi-shadow	Low
Cool	Low	Medium	Medium	Sunlight	Medium
Warm	High	Medium	High	Shadow	High
Hot	High	Low	Low	Shadow	High
Hot	Medium	High	Low	Shadow	High
:	:	:	:	:	:

NI LabVIEW is used to program the sensor nodes. LabVIEW is graphical programming environment used to develop measurement, test, and control systems using graphical codes. LabVIEW is integrated fully for communication with hardware such as GPIB, RS-485, and plug-in data acquisition boards. It uses terminology, icons, and ideas familiar to technicians, scientists, and engineers, and relies on graphical symbols rather than textual language to describe programming actions.

Fig. 6.6 depicts the Virtual Instrument (VI) block diagram. The VI block diagrams include data processing modules, and predefined mathematical models such as adders, substructures, integrators, etc. used in a mixed modality platform. In the block diagram, there is a terminal for every object created in the front panel.

At the beginning, the membership functions are set. For simplicity, only three linguistic variables for the input values “Low”, “Medium” and “High” are used. Afterwards, the output variable is determined based on the temperature, humidity, oxygen, and carbon dioxide values. In this code, the node will not transmit any radio messages until the probability of disease occurrence increase beyond 0.6.

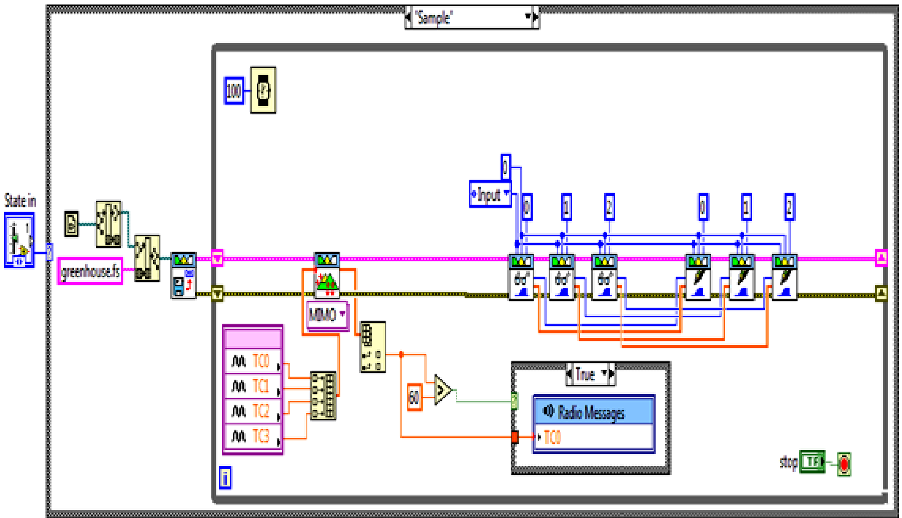


Fig. 6.6 Block diagram of the implementation of FLC in LabVIEW (VI)

6.5.2. GreenSense-2

Now, the second scheme will be considered in which the sensor nodes reports environmental parameters to the sink node. However, the fuzzy algorithm is implemented on the controller connected to the sink node. The FLC will be prototyped by using the Fuzzy Logic Toolbox in Matlab [35] as shown in Fig. 6.7.

Through this toolbox, prototype triangular fuzzy sets are set up for the fuzzy variables as shown in Fig. 6.4 and 6.5. The fuzzy inference consists of 243 fuzzy rules. The rule base is viewed graphically as shown in Fig. 6.8 in which different inputs can be chosen to see what the output will be.

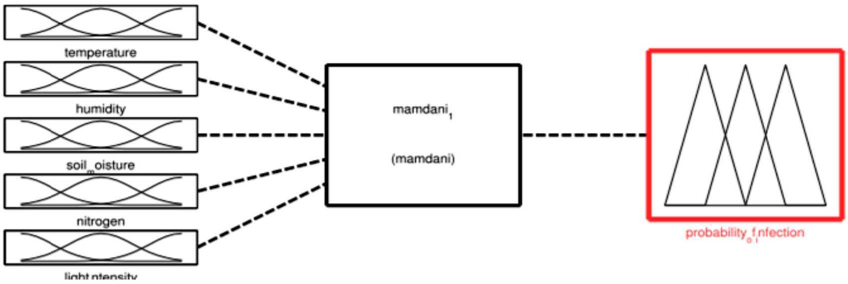


Fig. 6.7 Architecture of FLC controller in Matlab

As can be seen in Fig. 6.8, the first five columns on the left represent the input membership functions, the red line indicates the current input value. Whereas, this line crosses the membership functions are the values, which represent the degree to which the input is believed to belong to that particular membership. The sixth column on the right is the output membership functions. The output function on the bottom right represents the output of the FIS. The red bar on this graph indicates where the estimated value for the output variable lies.

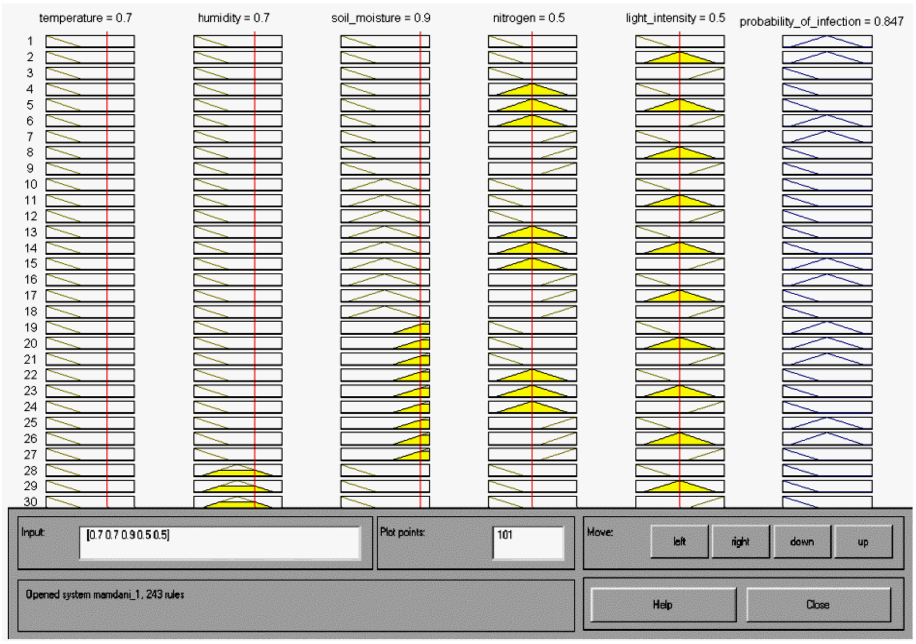


Fig. 6.8 Inferred output of the MISO system

Figs. 6.9 and 6.10 show examples for the control surface, which is the output, plotted against the two inputs, and the range of possible defuzzified values is displayed for all possible inputs. It is an interpolation of the effects of the entire rules. As shown in Fig. 6.9, the probability of infection increases as the humidity and temperature increases. While in Fig. 6.10, the probability of infection is small for normal level of nitrogen. However, this probability increases sharply when the level of nitrogen is very small or even very high.

GreenSense-2 is tested using a simple Simulink model as shown in Fig. 6.11. The same membership functions, fuzzy rules, and defuzzification method are utilized as those mentioned above. On the other hand, the inputs are simulated as proposed in [68]. For instance, temperature readings are simulated as a sine wave to present the day and night temperature changes.

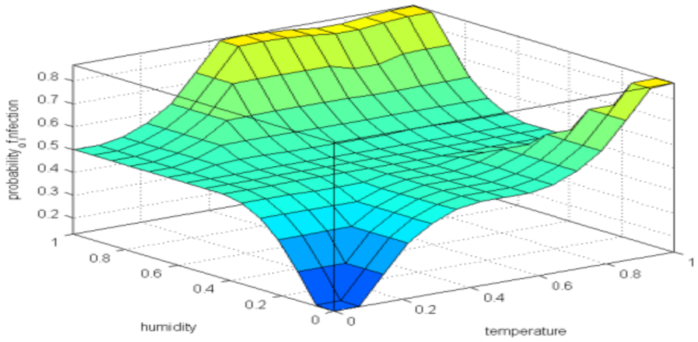


Fig. 6.9 Control surface for temperature and humidity

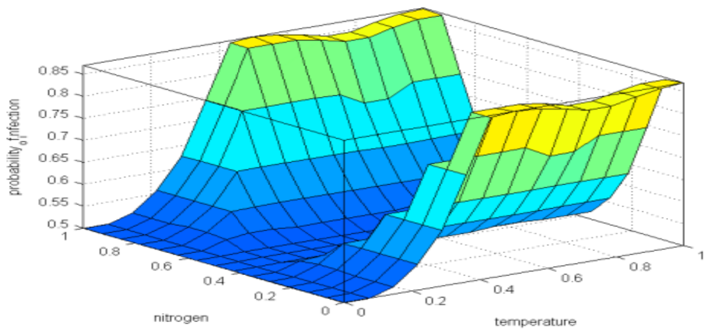


Fig. 6.10 Control surface for nitrogen and temperature

The amplitude of the wave is 5 degree plus a fixed bias of 30 degree with a frequency of 0.2618 rad/h. This frequency is measured according to a time period of 24 h. Fig. 6.12 shows the different inputs as sine waves with the same frequency where the wave in pink represents temperature, blue represents relative humidity, yellow is the soil moisture, red is the nitrogen level, and green for the solar radiation.

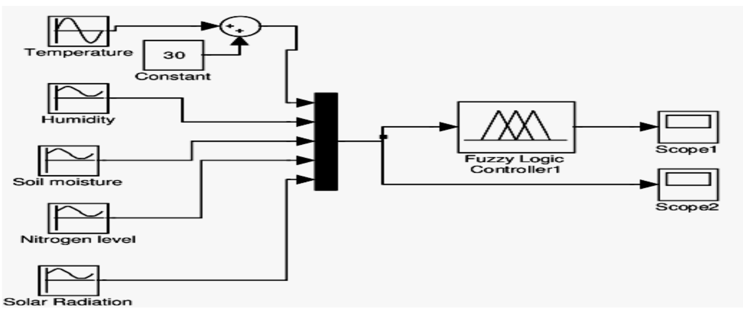


Fig. 6.11 Simulink model for fuzzy logic controller

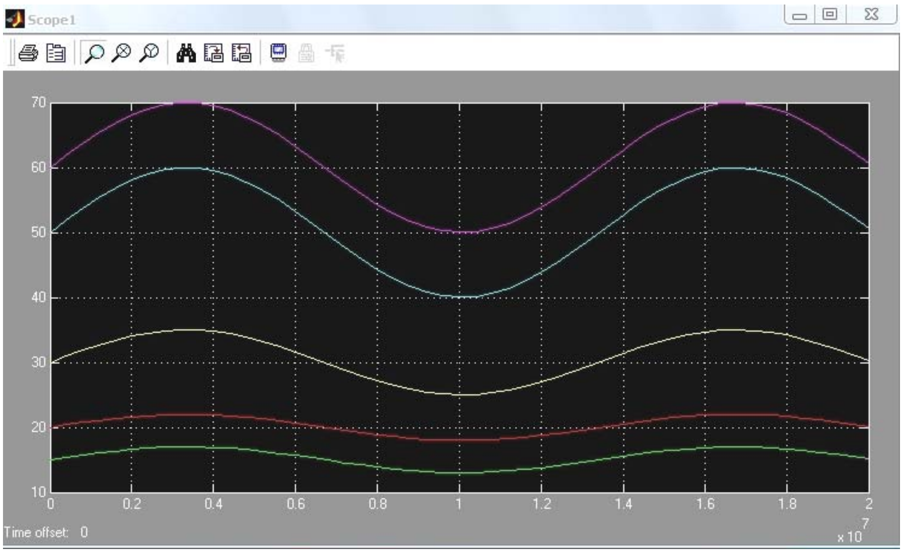


Fig. 6.12 The inputs to fuzzy logic controller

As can be seen in Fig. 6.13, the probability of infection is a pattern that is repeated every day according to the stimulus. These probabilities are generated based on the fuzzy rules. In addition, these rules are constrained by the suitable environmental conditions for the infection by a Gray leaf Spot disease as an example for illustration.

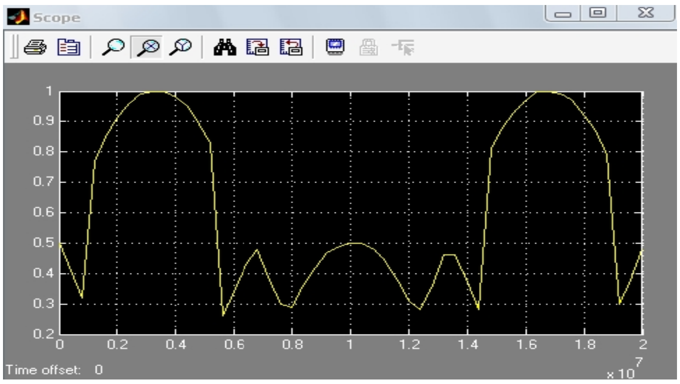


Fig. 6.13 Probabilities of infection of Gray Spot Leaf

The proposed FLC can be optimized through using the concept of Adaptive Neuro Fuzzy Inference System (ANFIS) [35]. The function of ANFIS is to allow the fuzzy system to learn by taking the FIS and tuning it with the back-propagation algorithm based on some collection of input-output data. In order to apply the previous FLC, we have to modify it according to the Sugeno inference algorithm instead of Mamdani method [11].

In the Sugeno implementation, the fuzzy model is modified by replacing the linguistic variables in the THEN part of the fuzzy rules with a crisp linear function of the input variables. The Sugeno rules have the following form:

$$R_i\colon if\ x_1\ is\ A_1^i\ and \ldots and\ x_n\ is\ A_n^i,$$

$$then\ y^i = p_0^i + p_x^i x_1 + \cdots + p_n^i x_n \tag{6.3}$$

With R_i is the rule numbered form 0 to i, x_j is the premise variable, A_j^i is the fuzzy subset defined by corresponding linear triangular membership function, and y^i is the consequent output of rule i. The output of the fuzzy system is determined as follow:

$$y = \frac{\sum_{i=1}^{N} w^i y^i}{\sum_{i=1}^{N} w^i} \tag{6.4}$$

Provided that: $$w^i = t_{j=1}^{n} A_j^i(x_j) \tag{6.5}$$

With t is the t-norm operator, the Sugeno implementation is much more efficient in terms of computation times because it is much simpler to determine the centroids of a two dimension shape. The main reason for employing ANFIS is the huge number of rules that will deteriorate the performance due to the computational overhead. ANFIS provide the option of generation a FIS with a smaller number of rules through subtractive clustering method. This method relies on partitioning the data into groups called clusters, and then create an FIS with the minimum number rules required to distinguish the fuzzy qualities associated with each of the clusters.

Fig. 6.14 shows the architecture of the ANFIS with subtractive clustering method where the input data are simulated using the approach proposed in [68]. Consequently, these input parameters are simulated as sine functions with very small frequency to follow the slow fluctuations during the day and the night.

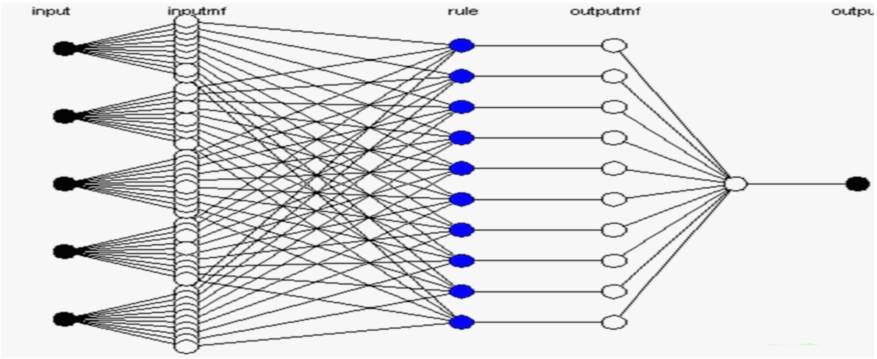


Fig. 6.14 ANFIS with Sub Cluster

The architecture is optimized to produce the fuzzy output through converting the values of the input variables into fuzzy variables where each node represents a membership function. Afterwards, the firing strength for each rule is estimated as shown in Fig. 6.15. Then, the output from each rule is calculated by the functions present in the THEN part of the rules. Finally, the system output is obtained by aggregating the outputs form the entire rules. The surface control for ANFIS with Sub Clustering is shown below in Fig. 6.16.

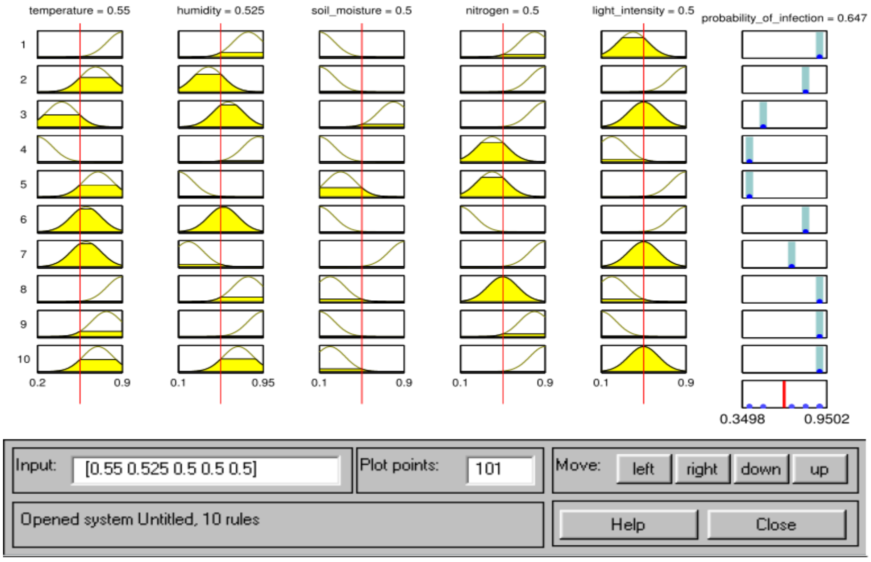


Fig. 6.15 The rules of ANFIS with Sub Clustering

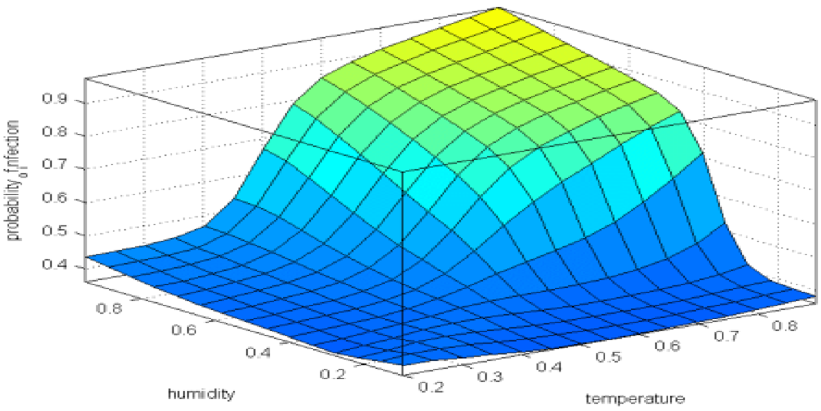


Fig. 6.16 The surface control for ANFIS with Sub Clustering

Chapter 7

Conclusion and Future Work

7.1. Conclusion

In this work, the overall lifetime of WSN is prolonged by minimizing the transmission among the sensor nodes. In addition, an application for the WSN in controlling the environmental conditions inside greenhouses is proposed.

The performance of our proposed approaches for maximizing the WSN lifetime was examined through simulations and practical experiments. The performance of our approaches was compared to the naïve model where each sensor sends its sensed data in a separate message to the sink node. The approaches utilized single and double exponential predictors, fuzzy logic algorithm, and threshold and tolerance e approaches to reduce the number of messages transmitted between the nodes and their neighbors as well as between the nodes and the sink node. Our approaches also targeted the reliability enhancement in sensors data. Therefore, the results of this work show that:

1. The proposed approaches would increase the lifetime of the network up to 100s' of times consequently the in-network traffic is minimized where the practical experiment confirmed the results obtained from the simulation.

2. Exponential Smoothing predictors can provide high performance even with randomly generated data. On the other hand, SES and DES predictors approximately achieves the same lifetime especially

with data with small trend such as temperature and humidity measurements.

3. Data reliability in case of DES is better than that in case of SES is used.

4. Fuzzy logic algorithm provides the maximum savings in the nodes' power. Therefore, multi-modal WSNs often consume the lowest power consequently achieving the highest lifetime.

5. The scalability of the network becomes limited due to the centralized model construction.

In addition, two prototypes, called GreenSense-1 and GreenSense-2, were proposed. Their idea is to help farmers detect plants diseases early using a network of sensor nodes with fuzzy logic controller as follow:

1. In GreenSense-1, fuzzy logic algorithm is implemented in sensor nodes to determine either the probability of infection for each disease.

2. In GreenSense-2, a network of sensor nodes are deployed where the function of these nodes is to report the environmental parameters without any internal processing except the REDR algorithm that runs for maximizing the lifetime of the deployed sensor nodes. This system is expected to work for several years by the dint of our proposed energy saving technique based on exponential smoothing predictors. Finally, the architecture of FLC was optimized using ANFIS technique to minimize the number of rules consequently improving the overall performance.

7.2. Future Work

In this section, the proposed modifications that may enhance our work and improve the performance of our proposed solutions are described. These modifications are listed below:

1. Sensors nodes that fit the improved features will be designed and fabricated where the sensor node' architecture will be implemented on FPGA in order to examine the performance of the dual prediction scheme when exponential smoothing predictors will be employed.

2. Other approaches to maximize the WSN lifetime will be investigated including data compression using wavelet transform, delta modulation, and differential PCM modulation.

3. Other predictors such as linear extrapolation, Markov algorithm, triple exponential smoothing, and artificial neural network based predictor will be examined.

4. GreenSense-1 and GreenSense-2 will be tested on a real greenhouse. In addition, these schemes will be improved by adding actuators that add more intelligence for the system and allow the operations without any human intervention.

5. Other applications for the WSN especially in the field of agriculture will be investigated in order to help farmers control their works efficiently.

References:

[1] A. Salhieh, J. Weinmann, M. Kochha, and L. Schwiebert, “Power efficient topologies for wireless sensor networks,” In Proc. of the IEEE International Conference on Parallel Processing, pages 156–163, 2001.

[2] A. Cerpa, D. Estrin, “Ascent: Adaptive Self-Configuring Sensor Network Topologies”, Proc. IEEE INFOCOM, 2002.

[3] A. Hernando, J. Ortega, J. Navarro, A. Prayati, and L. Lopez, “Problem Solving for Wireless Sensor Networks,” springer-Verlag London Ltd, 2008.

[4] A. Jain, E. Chang, and Y. Wang, “Adaptive Stream Resource Management Using Kalman Filters,” Proc. ACM International conf. on Management of Data (SIGMOD2004), Paris (France), pp. 11-22, June 13-18, 2004.

[5] Applications of fuzzy logic, http://www.jimbrule.com/fuzzytutorial.html, accessed Aug. 2011.

[6] B. Kanagal, and A. Deshpande, “Online Filtering, smoothing and Probabilistic Modeling of Streaming Data,” in Proc. of 24th International Conference on Data Engineering (ICDE 2008), 2008.

[7] B. Sundararaman, U. Buy, and D. Kshemkalyani, “Clock Synchronization for Wireless Sensor Networks: A Survey,” Ad Hoc Networks Elsevier, vol. 3, p. 281-323, 2005.

[8] B. Warneke, et al., "An Autonomous 16mm3 Solar-Powered Node for Distributed Wireless Sensor Networks," IEEE Sensors 2002, Orlando, FL, June 2002.

[9] Bluetooth, http://en.wikipedia.org/wiki/Bluetooth, accessed July 2011.

[10] C. Alippi and C. Galperti, “An Adaptive System for Optimal Solar Energy Harvesting in Wireless Sensor Network Nodes,” IEEE Trans. on Circuits and Systems, vol. 55, no. 6, pp. 1742–1750, July 2008.

[11] C. Leondes, “Fuzzy theory systems: techniques and applications”, vol. 1, Academic Press, Ch. 5, 1999.

[12] C. Mathuna, T. O’Donnell, R. Martinez-Catala, J. Rohan, and B. O’Flynn, “Energy scavenging for long-term deployable wireless sensor networks,” Talanta Special Section: Remote Sensin, vol. 75 (3), pp. 613-623, 2008.

[13] C. Schurgers, V. Tsiatsis, S. Ganeriwal, and M. Srivastava, “Optimizing sensor networks in the energy-latency-density design space,” IEEE Trans. on Mobile Computing, vol. 1, pp. 70 -80, Jan-Mar 2002.

[14] C. Shurgers, O. Aberthorne, M. Srivastava, "Modulation Scaling for Energy Aware Communication Systems," In Proc. of ISLPED 2001, pp 96–99, 2001.

[15] Carbon Dioxide, http://homeharvest.com/carbondioxideenrichment.htm, accessed June 2011.

[16] Coefficient of variation, http://en.wikipedia.org/wiki/Coefficient_of_variation, accessed Jan. 2011.

[17] Crossbow, Available: http://www.xbow.com/, accessed Jan. 2011.

[18] D. Chu, A. Deshpande, J.M. Hellerstein, and W. Hong, "Approximate Data Collection in Sensor Networks using Probabilistic Models," in Proc. 22nd International Conference on Data Engineering (ICDE06), pp. 48, Atlanta, GA, April 3-8, 2006.

[19] E. Breejen , M. Breuers, et al, "Autonomous forest fire detection," 3rd International Conference on Forest Fire Research and Fourteenth Conference on Fire and Forest Meteorology, Luso, Portugal, 1998.

[20] E. Gardner, "Exponential smoothing: The state of the art Part II," International Journal of Forecasting 22, 2006.

[21] Exploring the Protocol Stack of Wireless Sensor Networks, http://www.thecourse.us/Students/Wireless_Sensor_Networks.htm, accessed July 2011.

[22] Exponential Smoothing, http://en.wikipedia.org/wiki/Exponential_smoothing, accessed Jan. 2011.

[23] F. Alessandrini, M. Conte, S. Passerini, and P. Prosini, "Overview of ENEA's projects on lithium batteries," in Journal of Power Sources, vol. 97-98, pp. 768-71, 2000.

[24] F. L. Lewis, "Smart Environments: Technologies, Protocols, and Applications," Cook and S.K. Das, John Wiley, New York, 2004.

[25] F. Lafont, and J. Balmat, "Fuzzy Logic to The Identification and The Command of The Multidimensional Systems (Invited Paper)," published in the International Journal of Computational Cognition

(http://www.YangSky.com/yangijcc.htm) Vol. 2, No. 3, Pages 21–47, Sept. 2004.

[26] Fuzzy Control System, http://en.wikipedia.org/wiki/Fuzzy_control_system, accessed May 2011.

[27] G. Anastasi, M. Conti, M.D. Francesco, and A. Passarella, "Energy conservation in wireless sensor networks: A survey", presented at Ad Hoc Networks, pp.537-568, 2009.

[28] G. Mao, B. Fidan, and B. Anderson, "Wireless Sensor Networks Localization Techniques," Computer Networks, pp. 2529-2553, 2007.

[29] G. Pottie, W. Kaiser, "Wireless Integrated Network Sensors," Communication of ACM, vol. 43, N. 5, pp. 51-58, May 2000.

[30] G. Saridakis, D. Kolokotsa, and S. Dolianitis, "Development of an intelligent indoor environment and energy management system for greenhouses using a fuzzy logic controller and LonWorks® protocol," in Proc. of International Workshop on Energy Performance and Environmental Quality of Buildings, July 2006.

[31] Great Duck Island Project: http://www.coa.edu/motes.htm, accessed March 2011.

[32] H. Jung, and M. Pedram, "Stochastic Dynamic Thermal Management: A Markovian Decision-based Approach," International Conference on Computer Design, ICCD, 2006.

[33] H. Karl, "Quality of service in wireless sensor networks mechanisms for a new concept," ESF/PESC Exploratory Workshop on Wireless Sensor Networks, Zurich, April 1-2, 2004.

[34] H. Karl, and A. Willig, "Protocols and Architecture for Wireless Sensor Networks," 1st ed., John Willy and Sons. Ltd, pp. 167-194, 2005.

[35] H. Nguyen, N. Prasad, C. Walker, and E. Walker, "A First Course in Fuzzy and Neural Control," 1st edition, Chapman & Hall/CRC, 2003.

[36] H. Tekiner, and D. W. Coit, "System Reliability Optimization Considering Uncertainty: Minimization of the Coefficient of Variation for Series-Parallel Systems," in Proc. of the Industrial Engineering Research Conference (IERC), Vancouver, BC, May 2008.

[37] H. Yu Shwe, J. Xiao-hong, and S. Horiguchi: "Energy saving in wireless sensor networks," published at Journal of Communication and Computer, ISSN 1548-7709, USA, May 2009.

[38] *Handbook of Sensor Networks: Compact Wireless and Wired Sensing Systems*, 1st ed., CRC Press LLC, pp. 167-194, 2005.

[39] I. Hameed, "Simplified architecture of a type-2 fuzzy controller using four embedded type-1 fuzzy controllers and its application to a greenhouse climate control system," in Proc. of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering, 2009.

[40] I. Marín, E. Arceredillo, A. Zuloaga, and J. Arias, "Wireless Sensor Networks: A Survey on Ultra-Low Power-Aware Design" World Academy of Science, Engineering and Technology, 2005.

[41] I. Vasilescu, K. Kotay, D. Rus, M. Dunbabin, P. Corke, "Data collection, storage, retrieval with an underwater sensor network," in Proc. of the Third International Conference on Embedded Networked Sensor Systems (Sensys), San Diego, CA, 2005.

[42] Ian F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “A Survey on Sensor Networks,” IEEE Communications Magazine, August 2002.

[43] IEEE Standard 802.15.4, http://standards.ieee.org/about/get/802/802.15.html, accessed July 2011.

[44] Intel Lab Data, http://db.lcs.mit.edu/labdata/labdata.html, accessed Dec. 2010.

[45] J Zhang, P. Orlik, Z. Sahinoglu, A. Molisch, and P. Kinney, “UWB Systems for Wireless Sensor Networks,” in Proc. of IEEE, vol. 97, ISSN: 0018-9219, p. 313 – 331, 2009.

[46] J. Chang, and L. Tassiulas, “Maximum Lifetime Routing in Wireless Sensor Network,” appeared in IEEE/ACM Trans. on Networking, vol. 12, no. 4, pp. 609-619, 2004.

[47] J. A. Stankovic, et al, “Wireless Sensor Networks for In-Home Health-care: Potential and Challenges,” in High Confidence Medical Device Software and Systems (HCMDSS) Workshop, Philadelphia, PA, June 2-3, 2005.

[48] J. Elson and K. Romer, “Wireless Sensor Networks: A New Regime for Time Synchronization,” in Proc. of First Workshop on Hot Topics in Networks (HotNets-I), Princeton, New Jersey, 2002.

[49] J. Randall, “Designing Indoor Solar Products: Photovoltaic Technologies for AES,” John Wiley & Sons, ch. 5, 2005.

[50] J. Yick, B. Mukherjee, D. Ghosal, “Wireless sensor network survey,” Elsevier Computer Networks 52, pp. 2292–2330, 2008.

[51] K. Vijayaraghavan and R. Rajamani, “Active Control based Energy Harvesting for Battery-less Wireless Traffic Sensors: Theory and Experiments,” in Proc. of American Control Conference, NY, USA, pp. 4579–4584, July 11-13 2007.

[52] K. Kalpakis, K. Dasgupta, and P. Namjoshi, “Maximum Lifetime Data Gathering and Aggregation in Wireless Sensor Networks,” appeared in Proc. of the 2002 IEEE international conference on networking (ICN), pp. 685–696, 2002.

[53] K. Maraiya, k. Kant, and N. Gupta, “Application based Study on Wireless Sensor Network,” appeared in the International Journal of Computer Applications (0975 – 8887), vol. 21– no.8, May 2011.

[54] L. Bernardo, R. Oliveria, R. Tiago, P. Pinto, “A Fire Monitoring Application for Scattered Wireless Sensor Network,” in Proc. of WinSys 2007, on July 28-31 2007.

[55] L. Hu, and D. Evans, “Localization for Mobile Sensor Networks,” in Proc. of 10th International Conference on Mobile Computing and Networking (MobiCom 2004), Philadelphia, 45-47, 2004.

[56] M. Bahrepour, N. Meratnia, and P. Havinga, “Automatic Fire Detection: A Survey from Wireless Sensor Network Perspective,” Technical Report, TR-CTIT-08-73, Centre for Telematics and Information Technology University of Twente, Enschede, ISSN 1381-3625, 2008.

[57] M. El Aoud, M. Franceschi, and M. Maher, “Self – Tuning Method of Fuzzy System: An Application on Greenhouse Process,” in Proc. of World Academy Of Science, Engineering And Technology, vol. 25, Nov. 2007.

[58] M. Panoiu, C. Panoiu, R. Rob, and L. Ghiorghioni, “System based on Fuzzy Logic for maintain Optimum Environmental Conditions in a Fir Tree Greenhouse,” in Proceeding ICS'10 in Proc. of the 14th WSEAS international conference on Systems: part of the 14th WSEAS CSCC multi-conference, 2010.

[59] M. Zorzi, and R. Rao, “Geographic Random Forwarding (GeRaF) for ad hoc and sensor networks: multihop performance,” appeared in the IEEE Trans. on Mobile Computing, vol. 2, n. 4, Oct.-Dec. 2003.

[60] N. P. Borg, and C. Debono, “An FPGA implementation of adaptive data reduction technique for WSN,” presented in Proc. of WICT 08 Conference, 2008.

[61] NI WSN architecture: http://zone.ni.com/devzone/cda/tut/p/id/11211, accessed May 2011.

[62] NI WSN Starter kit: http://sine.ni.com/nips/cds/view/p/lang/en/nid/206916, accessed May 2011.

[63] Nokia's Wibree and the Wireless Zoo,

http://thefutureofthings.com/articles.php?itemId=38/59/, accessed July 2011.

[64] Ohio State University, "Recommended Storage Temperature and Relative Humidity Compatibility Groups Report," found at
http://ohioline.osu.edu/fresh/Storage.pdf, accessed May 2011.

[65] Ornamental Diseases,

http://www.greenhousemanagementonline.com/greehhouse-0211-recognizing-disease-symptoms-ornamentals.aspx, accessed June 2011.

[66] P. B. Godfrey, D. Ratajczak, “Naps: Scalable, Robust Topology Management in Wireless Ad Hoc Networks”, in Proc. of the 3rd international Symposium on information Processing in Sensor Networks (IPSN ‘04), Berkeley, California, USA, April 26-27, 2004.

[67] P. Honeine, C. Richard, M. Essoloh, and H. Snoussi, “Localization in Sensor Networks – A Matrix Regression Approach,” in proc. of Sensor Array and Multichannel Signal Processing Workshop, 2008.

[68] P. Javadi Kia, A. Tabatabaee Far, M. Omid, R. Alimardani and L. Naderloo, “Intelligent Control Based Fuzzy Logic for Automation of Greenhouse Irrigation

System and Evaluation in Relation to Conventional Systems," published in World Applied Sciences Journal 6: 16-23, ISSN 1818-4952, 2009.

[69] P. Lanfang, W. Wanliang, and W. Qidi, "Application of Adaptive Fuzzy Logic System to Model for Greenhouse Climate," in Proc. of the 3d World Congress on Intelligent Control and Automation, June 28-July 2, 2000.

[70] P. Ranganathan, and K. Nygrad, "Time synchronization in Wireless Sensor Network: A Survey," International journal of Ubicomp (IJU), vol. 1, no. 2, 2010.

[71] P. Santi "Topology Control in Wireless Ad Hoc and Sensor Networks," John Wiley & Sons, Ltd, 2005.

[72] P. Zhang, C.M. Sadler, S.A. Lyon, M. Martonosi, "Hardware design experiences in ZebraNet," in Proc. of the SenSys'04, Baltimore, MD, 2004.

[73] Predictor's Weights, http://vbautomation.110mb.com/FORECASTING/expoential_smoothing_method.htm, accessed Jan. 2011.

[74] Q. Han, S. Mehrotra, N. Venkatasubramanian, "Energy Efficient Data Collection in Distributed Sensor Environments", in Proc. of the 24th IEEE International Conference on Distributed Computing Systems (ICDCS'04), pp. 590-597, March, 2004.

[75] R. Wattenhofer, L. Li, P. Bahl, and Y. M. Wang, "Distributed Topology Control for Power Efficient Operation in Multihop Wireless Ad hoc Networks," in Proc. of IEEE INFOCOM, 3:1388–1397, April 2001.

[76] R. Casas, H. Gracia, A Marco, and J. Falcó, "Synchronization in Wireless Sensor Networks Using Bluetooth," in Proc. of Third International workshop on Intelligent Solutions in Embedded Systems, 2005.

[77] R. Frank, "Understanding Smart Sensors," second Ed., Artech House, Norwood, MA, 2000.

[78] R. Verdone, D. Dardari, G. Mazzini, and A. Conti, "Wireless sensor and actuator network technologies: analysis and design," AP LCC, 2009.

[79] R.J. Bogers, L.E. Craker, and D. Lange (eds.), "Medicinal and Aromatic Plants," 1st ed., Springer, ch. 17, pp. 237-252, 2006.

[80] S. Ekern, "Adaptive Exponential Smoothing Revisited," Journal of the Operational Research Society, vol. 32 pp. 775−782, 1982.

[81] S. Goel and T. Imielinski, "Prediction-based Monitoring in Sensor Networks: Taking Lessons from MPEG", ACM Computer Communication Review, vol. 31, no. 5, October 2001.

[82] S. Goel, A. Passarella and T. Imielinski, "Using buddies to live longer in a boring world", in Proc. IEEE International Workshop on Sensor Networks and Systems for Pervasive Computing (PerSeNS 2006), Pisa (Italy), March 13, 2006.

[83] S. Lindsey, C. Raghavendra, and K. Sivalingam, "Data Gathering in Sensor Networks using the Energy Delay Metric," appeared in IEEE Trans. on Parallel and Distributed Systems, 13(9), 924–935., 2001.

[84] S. Misra, I. Woungang, S. C. Misra, "Guide to Wireless Sensor Networks," Springer-Verlag London Limited, ch. 17, 2009.

[85] S. Pyo, and Y. Choi, "A Fast Handover Scheme Using Exponential Smoothing Method," appeared in IJCSNS International Journal of Computer Science and Network Security, vol. 9 No.2, Feb. 2009.

[86] S. Roundy, "Energy Scavenging for Wireless Sensor Nodes with a Focus on Vibration-to-Electricity Conversion," presented at BWRC Winter 2003 Retreat, 2003.

[87] S. Santini, and K. Römer, "An Adaptive Strategy for Quality-Based Data Reduction in Wireless Sensor Networks," in Proc. of the third International Conference on Networked Sensing Systems (INSS 2006), TRF, pp. 29-36, Chicago, IL, USA, June 2006.

[88] S. Wyne, J. Karedal, P. Almers, F. Tufvesson, and A. Molisch, "UWB Channel Measurements in an Industrial Environment," in Proc. of Global Telecommunications Conference - GLOBECOM, 2009.

[89] T. Ross, "Fuzzy logic with engineering applications," 1st edition, John Wiley & Sons Ltd, 2004.

[90] T. S. Rappaport. "Wireless communications: Principles and Practice," 1st ed., Prentice Hall, 1996.

[91] The Editors, "10 emerging technologies that will change the world: Technology Review", vol. 106, no.1, pp. 33-49, Feb. 2003.

[92] Time series analysis, http://www.statsoft.com/textbook/time-series-analysis/#simple, accessed Aug. 2011.

[93] V. Raghunathan, C. Schurghers, S. Park, M. Srivastava, "Energy-aware Wireless Microsensor Networks", IEEE Signal Processing Magazine, , pp. 40-50, March 2002.

[94] V. kumar, S. verma, and S. L. Rao, "Low Power Wireless Sensor Network," Seminar report, Moradabad Institute of Technology, 2009.

[95] V. Shnayder, M. Hempstead, B. Rong Chen, G. Werner-Allen, and M. Welsh, "Simulating the power consumption of large-scale sensor network applications," in Proc. of the Second ACM Conference on Embedded Networked Systems (SenSys), Nov. 2004.

[96] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, “Energy-Efficient Communication Protocol for Wireless Microsensor Networks,” in Proc. 33rd Hawaii International Conference on System Sciences-vol. 8, p.8020, Jan. 04-07, 2000.

[97] W. Heinzelman, J. Kulik, H. Balakrishnan, “Adaptive protocols for information dissemination in wireless sensor networks,” in Proc. of the ACM MobiCom’99, Seattle, Washington, pp. 174–185, 1999.

[98] W. Seah, Z. Ang Eu, and H. Tan, “Wireless Sensor Networks Powered by Ambient Energy Harvesting (WSN-HEAP) – Survey and Challenges,” in Proc. of 1st International Conference on Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronics Systems Technology, 2009.

[99] Wavenis: http://en.wikipedia.org/wiki/Wavenis, accessed July 2011.

[100]Wikipedia, “Wibree,” http://en.wikipedia.org/wiki/Wibree, accessed July 2011.

[101] Wireless Sensor and Actor Networks, http://www.ece.gatech.edu/research/labs/bwn/actors/work.html, accessed July 2011.

[102] WSN Simulator, http://www.djstein.com/projects/WirelessSensorNetworkSimulator.html, accessed April 2011.

[103] X. Zhang, G.F. Riley, “Bluetooth Simulations for Wireless Sensor Networks using GTNetS,” 12th IEEE/ACM International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS), Oct. 2004.

[104] Y. Xu, J. Heidemann, and D. Estrin, “Geography-informed Energy Conservation for Ad Hoc,” in Proc. ACM MobiCom, 2001.

[105] Y. Kwon, K. Mechitov, S. Sundresh, W. Kim, and G. Agha, “Resilient Localization for Sensor Networks in Outdoor Environments,” In Proc. of 25th IEEE International Conference on Distributed Computing Systems (ICDCS), 643-652, 2005.

[106] Zigbee Standards Overview, http://www.freescale.com/webapp/sps/site/overview.jsp?nodeId=01J4Fs25657725, accessed July 2011.

[107] Zigbee, http://en.wikipedia.org/wiki/ZigBee, accessed July 2011.

Publications

[1] M. Abdel-Aal, R. Ramadan, Ahmed A. Dessouki, and M. Abdel-Meguid, "**An Efficient Data Reduction Technique for Intelligent Environments**," accepted at the Intelligent Environment Conference (IC'11), Nottingham Trent University, United Kingdom, 2011.

[2] M. Abdel-Aal, R. Ramadan, Ahmed A. Dessouki, and M. Abdel-Meguid, "**Reliable and Efficient Data Reduction Techniques for Single and Multi-Modal WSNs**," submitted to the Computer Communication Journal, Special Issue on Reactive Sensor Networks, 2011.

[3] M. Abdel-Aal, R. Ramadan, Ahmed A. Dessouki, and M. Abdel-Meguid, "**GreenSense: Towards Utilization of Wireless Sensor Networks in Greenhouses Applications**," submitted to the Journal of Ambient Intelligence and Smart Environments, 2011.