



BUDAPEST UNIVERSITY OF TECHNOLOGY AND ECONOMICS
Department of Networked Systems and Services

Innovative techniques to robust wireless sensors networks

Ph.D. dissertation

Moahmmed Al-Mazaideh

supervised by
Dr. Levendovszky János, D.Sc.
Doctor of the Hungarian Academy of Sciences

January 21, 2022

Acknowledgements

First and foremost, I would like to thank my supervisor Prof. Janos Levendovszky; I am deeply indebted to him for advising me honestly through my journey of Ph.D. at Budapest University of Technology and Economics. His immense knowledge and experience were my guide to my scientific and academic achievements.

I also want to thank Tempus foundation for funding my scholarship via the Stipendium Hungaricum program.

I would also like to thank all members of the Networked Systems and Services Department and BME staff for the helpful support I had during my study period.

I am very thankful to my colleagues and my friends, especially Mr. Dhari Ali and Mr. Yehia al Naiemy and their families, for the appreciated patronage I had.

At last, I am so grateful to my extended family, my late father, mother, brothers, and sisters; they are the roots of my being and humanity. My most tremendous gratitude to my exceptional wife, Abeer; I am beholden to her; her patience, love, and inspiration are the candles that light up my life. Big hugs and love to my kids, the hope, and the extension of my existence.

Executive summary

Recently, Wireless Sensor Networks (WSNs)- as a subset of IoT systems - have become the backbone of several applications targeting different aspects of data acquisition. This fact promoted researchers to develop further technologies enhancing the performance of WSN. Expansion of applications and other necessities, besides developing co-operating technologies (such as communication, electronics, real-time embedded systems,etc.), entails the challenge of further improving the performance of WSNs. In this dissertation, I proposed different techniques that improve the performance of WSNs, in terms of energy efficiency, reliability, throughput, and delay. These results can contribute to achieving a better quality of service in real-time data collection..

In the first thesis[1] , I developed two novel energy-aware and reliable routing protocols. The aim was to maximize the lifespan of wireless sensor networks (WSNs) subject to predefined reliability constraints by using multi-hop routing schemes. The source node forwards the packet to the Base Station (BS) via other nodes as relays. In the first proposed protocol, energy efficiency is achieved by maximizing the minimum residual energy of the path subject to fulfilling a predefined reliability constraints. The second protocol is an optimized version of the first one with respect to lifespan and complexity. The optimal path is the one in which the residual energy distribution of the nodes along the path is as close to uniform as possible and the packet arrives at the BS with a given success probability. To measure the uniformity of the residual energy distribution, I used an entropy like measure. The information about the current energy state of the network is maintained by using a look-up-table from which the optimal routes are computed on the BS. The BS broadcasts the updated optimal paths to each node after each round of packet transmission.

The deterministic WSNs covered in the first thesis were extended in the second thesis [2] to include stochastic WSNs where packets are generated, stored and transmitted subject to queuing dynamics. I presented a new energy-efficient and reliable routing protocol for WSNs, including a stochastic traffic generation model and a sleep/active (ON/OFF) mechanism. My objective was to improve the longevity of the WSNs by energy balancing but providing reliable packet transfer to the Base Station at the same time. The proposed protocol is based on the principle of the back-pressure method, besides the difference of backlogs, to optimize energy consumption, I used a cost function related to an entropy-like function defined over the residual energies of the nodes. In the case of two-hop routing, the optimal relay node is selected as

the one which has maximum backlog difference and keeps the distribution of residual energy as close to uniform as possible. Uniformity is measured throughout the network by the change of the entropy of the residual energy of the nodes. The protocol assumes a Rayleigh fading model. Simulation results show that the proposed algorithm significantly improves the performance of traditional back-pressure protocol with respect to energy efficiency, E2E delay, and throughput, respectively.

In the third thesis [3], I developed a data reduction technique to improve the energy efficiency and reliability of WSNs. Compressive sensing (CS) is a data reduction technique used to recover extensive data from fewer samples in case of sparse representation of sensor readings. Unfortunately, energy efficiency and accuracy are contradictory metrics, as increased accuracy requires a large number of measurements and thus consequent data transmissions. Therefore, a CS-based algorithm was proposed for efficient data transfer through WSNs, which uses a multiple objectives genetic algorithm (MOGA) to optimize the number of measurements, transmission range, and the sensing matrix. The algorithm aims at striking a good balance between energy efficiency and accuracy. It constructs a path in a multi-hop manner based on the optimized values. Numerical simulations and experiments show that Pareto-front, which is the output of MOGA, helps the user to select the right combination of the number of measurements and the transmission range fitting the application at hand and to strike a good balance between energy efficiency and accuracy. The results also demonstrate the existence of measurement matrices that have lower mutual coherency improves the accuracy of CS.

In chapter (5), I proposed a predictive maintenance system (PdM) utilizes Machine Learning (ML) approach to predict the system's operational status after M active steps based on L previous observations, ML approach was implemented by a Feedforward Neural Network (FFNN). I applied the proposed model to build a PdM system for WSNs, where my concern was to predict the state of the system as far as the quality of data transfer is concerned. The *FFNN* provides a forward prediction of the operational status of the network after M consecutive time steps in the future, based on the previous L readings of QoS requirements of WSN (data loss, energy efficiency, throughput, and delay). I used quantization and encoding schemes to reduce the complexity of the system which enables easy software implementation for limited resources WSNs. I also demonstrate the relation between complexity and accuracy (L and M , respectively). I found that larger M leads to longer execution time and larger prediction error, where larger L entails longer execution time and smaller prediction error. I also investigate how quantization and encoding can reduce complexity to implement a real-time PdM.

Contents

<i>Acknowledgments</i>	i
<i>Executive summery</i>	ii
<i>List of figures</i>	vi
<i>List of tables</i>	viii
1 Introduction	1
1.1 Reliability and energy efficiency in deterministic WSN	3
1.1.1 Problem statement	3
1.1.2 Objectives	3
1.2 Throughput maximization in stochastic bursty WSN	4
1.2.1 Problem statement	4
1.2.2 Objectives	5
1.3 Energy efficiency based on data reduction	5
1.3.1 Problem statement	5
1.3.2 Objectives	6
1.4 Predictive Maintenance systems	6
1.4.1 Problem statement	7
1.4.2 Objectives	7
1.5 The structure of the dissertation	8
2 Novel reliable and energy-efficient routing protocols for WSN	9
2.1 Introduction	9
2.2 Related Work	11
2.3 The model	13
2.4 Routing to Maximiz the Minimal Residual Energy	14
2.4.1 2-hop routing	14
2.4.2 Extension to m-hop routing	16
2.5 Optimal path selection based on networks state energy entropy	18
2.6 Simulation results	20
2.7 Conclusions	25

3	<i>A new energy-efficient and reliable protocol for stochastic WSNs</i>	27
3.1	Introduction	27
3.2	Related work	29
3.3	The system model	30
3.4	Routing with reliability and energy balancing	32
3.5	Simulation and performance evaluation	34
3.6	Conclusions	38
4	<i>Reliable and energy-efficient algorithm based on data reduction</i>	39
4.1	Introduction.	39
4.2	Related works.	41
4.3	Compressive sensing in WSNs.	42
4.4	MOGA in WSNs.	44
4.5	CCS-MOGA	47
4.6	Simulation and numerical results	55
4.7	Conclusions	60
5	<i>Predictive maintenance for WSN</i>	62
5.1	Introduction	62
5.2	Related work	64
5.3	The system model	65
5.3.1	Predicting the forward probability distribution by FFNN	66
5.4	”Customized” FFNN for implementing PdM for WSNs – quantization and sparsity	68
5.5	Dataset for training	70
5.6	Implementation and simulation results	73
5.7	Conclusions	76
6	<i>Summary of results and thesis</i>	77
6.1	Summary of results	77
6.2	The novel scientific results of the dissertation	78
6.3	Possible future works	79
	<i>Publications</i>	81
	<i>Bibliography</i>	81

List of Figures

2.1	The flowchart of MMREP	17
2.2	The flowchart of OREBP	20
2.3	Simulated Network	21
2.4	The network as a graph	22
2.5	Distribution of Residual Energy of the path at the moment when the first node goes flat.	23
2.6	Lifespan with different data loss (ε).	23
2.7	Success probability when data loss ($\varepsilon = 0.05$).	24
2.8	lifespan & maximum number of hops ($\varepsilon = 0.05$).	24
2.9	Lifespan, Number of hops and Execution time.	25
3.1	The node as a queue	31
3.2	The variance of consumed energy VS length of the chain	35
3.3	Total consumed energy vs length of the chain	35
3.4	Variance of consumed energy vs Utilization ρ	36
3.5	Total consumed energy vs Utilization ρ	36
3.6	Throughput vs packet generation rate γ	37
3.7	Lifespan vs Utilization ρ	37
3.8	Average delay time vs Utilization ρ	38
4.1	Pareto-front.	45
4.2	Pareto-optimal front.	48
4.3	Distribution of leaf-nodes.	50
4.4	Shortest paths.	51
4.5	Compressive sensing model.	52
4.6	The flowchart of CCS-MOGA	53
4.7	The flowchart of CS algorithm in the BS side	54
4.8	The flowchart of CS algorithm in CL side	55
4.9	Coherency of random and optimal Φ	56
4.10	Reconstruction error of optimal Φ and random Φ	56
4.11	The reconstruction error per transmission round.	57
4.12	The average consumed energy per transmission round.	57
4.13	The relation among Energy, error and number of measurements of a WSN consists of 500 nodes.	58
4.14	The relation among energy, error and transmission rang of WSN consists of 500 nodes.	58

4.15	The average consumed energy per node.	59
4.16	The variance of the consumed energy.	60
5.1	FFNN of three hidden layers, and $K=8$ previous observations window.	68
5.2	Samples of the input vector as black and white patterns.	74
5.3	Complexity of original data and quantized and encoded data . . .	74
5.4	Accuracy of original data and quantized and encoded data . . .	75
5.5	The relation among MSE , Execution time, and M	76
5.6	The relation among MSE , Execution time, and K	76

List of Tables

4.1	Pareto-optimal objectives and variables.	48
5.1	Pre-configured parameters	70
5.2	Measured parameters	71
5.3	Sample of Observations table	71
5.4	Sample of the training dataset	73
5.5	Sample of the dataset after quantization and encoding.	73

Chapter 1

Introduction

Wireless sensor networks are distributed systems consisting of a base station and a number of nodes. Each node is equipped with application-specific sensors, a simple processing unit, a wireless radio transceiver, and a power unit. These types of equipment enable the node to capture and collect data, process it, discover its neighborhood, and communicate with each other and with the BS. The wide variety of sensors that can be used in WSNs (e.g., optical, thermal, radar, bio, and acoustic sensors) make it possible to use WSNs in any applications that need extensive data accession, such as military applications, bio-medical applications, transportation applications, environmental applications, smart appliance applications; besides that, the modern IoT applications form a major evolution of WSNs.[4].

The implementation of WSNs usually follows the IEEE 802.15.4 standards which define the specifications of protocols that can be used in low-power, low-cost and low-rate wireless networks in physical and MAC layers [5]. IEEE 802.15.4 supports wireless networks that consist of at least one Full Function Device (FFD) and other Reduced Function Devices (RFD) networked together in star, peer to peer or clustered tree topology. It supports 20,40 and 250 kbps data rates. These standards support Carrier Sense Multiple Access / Contention Avoidance (CSMA/CA). The typical life of battery power strongly depends on the activity cycle of the device (active mode/sleep mode) and the number of transmitted packets. The applied communication protocol can also control the energy used for packet transmission on each node, as a result there is a need for low energy information transmission.

Designing and operating of WSN should respect some constrains such as:

- Limited resources: The nodes are cheap and low quality, so they have limited memory, processing capitalises and communication bandwidth.
- Limited energy: The nodes are typically battery operated. Hence, energy becomes a scarce resource.
- Reliability of communications: WSN usually operated in harsh environments in terms of communications, nodes transmits data to each other and to the BS via unreliable radio channels corrupted by noise and fading.

- Dynamic topology: The topology of WSN changes frequently, some new nodes may be added or returned to service after a pause, and some nodes may be faded by running out of energy, failure or malicious attack.

Despite these limitations, the performance of any system is supposed to satisfy some minimum level of services and requirements, which are known as Quality-of-Service (QoS)[6], in case of WSN, QoS include reliability, energy efficiency, security, accuracy, delay and-so-forth. Besides QoS, lifespan is a curial measure of robustness of WSN. Researchers use several definitions of lifespan, some of them based on number of alive nodes, in this case, the lifespan is the time needed for n numbers of the nodes to go flat, other definition based on coverage, network connectivity or application requirements.

Many researchers suggest various strategies and techniques to guarantee QoS and to prolong the lifespan of WSNs. These strategies and techniques cover almost all the layers of networking protocols ranging from the physical layer to the application layer; they include data reduction, protocol overhead reduction, energy-efficient routing protocols, duty cycling, topology and mobility, and energy harvesting. Some of these requirements and constraints contradict; hence the objective is often to strike the right balance among them, which proves to be a major challenge[7].

In this research, I aimed to present novel solutions to improve the efficiency of WSNs in terms of energy-efficient routing protocols, data reduction, and data quality. The proposed solutions cover the following aspects of WSN designing and operating, energy-efficient routing algorithm for deterministic WSN, energy-efficient routing algorithm for stochastic WSN, data reduction algorithm based on compressive sensing paradigm, and predictive maintenance system for WSNs. All the proposed algorithms assume that all the nodes are battery operated and the base station is rigged up with a wired electrical power source.

The rest of this chapter is organized as follow:

- In Section 1.2, I provide an overview, problem statement, and my objectives in the area of reliability and energy efficiency.
- Section 1.3 introduces an overview, the problem statement, and my research objectives in routing efficient and reliable protocols for stochastic WSNs.
- In Section 1.4, I present an overview of data reduction techniques in WSNs; I introduce the problems statement and my objective of introducing new routing algorithms based on data reduction.
- In Section 1.5, I describe PdMs systems and the problem statement of predictive maintenance systems of WSNs briefly; I also state the objectives of my proposed PdM system.
- In Section 1.6, I describe the structure of my dissertation.

1.1 Reliability and energy efficiency in deterministic WSN

1.1.1 Problem statement

In deterministic WSN, the nodes transmit the data packets in a predefined schedule, as aforementioned nodes are typically small and battery-operated, and the data is transmitted to the BS over unreliable radio channels corrupted by noise and fading. Energy efficiency is not only concerned about the amount of consumed energy but for the sake of network longevity; it is also about guaranteeing uniform energy distribution among the nodes. The lack of load balancing may cause the early death of some nodes, which in turn - breaks the connectivity of WSN and reduces the life span of the whole network. Besides energy efficiency, reliability is a curial concern in WSNs. Low energy and unreliable transmissions can cause a high packet loss ratio. Unfortunately, both energy efficiency and reliability as QoS requirements conflict with each other.

Most protocols (LEACH, PEDAP, PEGASIS) use techniques such as clustering and chaining to achieve energy efficiency by enforcing uniform energy distribution and maximum longevity, but they do not provide reliable transmissions, whereas lost data means wasted energy. Such protocols and their modifications use methods like round-robin, and they try to optimize the remaining energy for repeated selection of different cluster heads and chain leaders. However, they do not achieve uniformity of energy distribution and load balancing. In the absence of such optimization, the selection for the cluster heads or the chain leaders is inefficient, and they can become bottlenecks of the WSN.

A lot of researches as [8, 9, 10] improve reliability either by providing more redundant data, which increases energy consumption, or by providing alternative links and backup devices, which increases the cost and is incoherent with the limited capacity of nodes. Such a contradiction between energy efficiency and reliability makes it imperative to look for new solutions that optimize both and provide an accurate tool to estimate the uniformity of energy distribution.

1.1.2 Objectives

My objective in the area of reliability and energy efficiency was to develop new algorithms for WSNs that maximize network longevity (achieve maximum throughput) under the constraint of reliable communication, the proposed algorithms should also, select the optimal path that guarantees a uniform energy distribution based on clear and proved measure.

The key contributions of this field are summarized as follows:

- I presented a mathematical analysis for the selection of relay nodes based on the remaining energy and reliability.

- I proposed Maximum of the Minimum Residual Energy Protocol (MM-REP) to select the path that maximizes the remaining energy of its nodes subject to the reliability constraint.
- I introduced the entropy-like function to measure the uniformity of residual energy distribution.
- I proposed Optimal Remaining Energy-Based Protocol (OREBP), which selects the path based on the entropy of residual energy of the path.

The proposed algorithms can be used not only in WSN, but also any routing-based systems such as telecommunication, transportation, they also can be used in the systems need load balance such as resources management systems.

1.2 Throughput maximization in stochastic bursty WSN

1.2.1 Problem statement

In stochastic WSN, packets are generated and transmitted subject to queuing dynamics. Such a scheme of data transmission is used as an energy-efficient technique known as the ON-OFF(sleep/wakeup) scheme, some WSN applications require this paradigm, such as surveillance and event-based applications. In stochastic WSNs, MAC manages to build in queues-like where the data is held prior to transmission.

Considering the nodes as queues requires the designer to be aware of influential issues, first; the limited storage space of the node, as previously stated; the simplicity and low quality of sensor node lead to the smallness of available buffering space. Second; End-to-end delay; where in queuing theory, the expected time a packet resides in the queue depends on the expected queue length, packet blocking probability, arrival rate, and servicing rate. Third; Congestion occurrence which depends on the queue length and the queue length fluctuations. Forth; Energy efficiency; the designer should provide a mechanism to control the ON-OFF scheme in a way that guarantees the balance between energy efficiency and other issues like delay and buffering overload. Fifth, reliability, short queues, slow service rate, and high arrival rate increase the probability of lost packets which means low reliability of the WSN.

The backpressure principle is used widely in designing of routing protocols for stochastic WSNs, where the movement of the packets among the nodes depends mainly on the backlog difference of the nodes. Researches like [11, 12] uses the backpressure principle in a way that improves End-to-End delay and complexity but decreases energy efficiency, where [13, 14] uses it in a form improves the energy efficiency, but in term of the amount rather than load-balance. We still need solutions that strike the optimal load balance in terms

of energy distribution, reliability and queue length. The systems using the ON-OFF scheme should not cause a bursty system that may affect negatively the QoS requirements in terms of reliability and delay.

1.2.2 Objectives

The objective of my research in the field of stochastic energy-efficient routing was to develop a new algorithm that provides a good balance among QoS requirements of WSNs, such as reliability, energy efficiency, throughput and delay. The proposed solution should reduce the concerns about the appearance of some phenomena associated with stochastic WSNs such as end-to-end delay and low reliability due to overflowing and congestion. Energy-efficiency should be maintained in both dimensions; the amount and the distribution.

In this research, I proposed an algorithm that maintains balance and stable stochastic WSN in terms of energy-efficiency, reliability, throughput and delay. The proposed solution combines four significant principles;

- The activities of the nodes are controlled by On-OFF scheme to provide energy-efficiency.
- Back-pressure principle is used to control the traffic load balance to eliminate congestion.
- Entropy of residual energy is used as measure to control the uniform distribution of energy.
- Rayleigh fading model is proposed to balance between consumed energy and reliability.

Besides WSNs; the proposed algorithm can be used in routing based systems such as telecommunication, transportation, it also can be used in the systems use queueing theory to manage resources.

1.3 Energy efficiency based on data reduction

1.3.1 Problem statement

As aforementioned, energy efficiency is a curial concern in designing and operating of WSN. nodes expend Most the energy in communication activities with each other or with the BS, which makes it important to save the energy from wasting in transmission and reception of useless data, and it's of wisdom to utilize the minimum amount of data that can provide the message. Data reduction techniques are effective tools used not only to improve the energy efficiency but also reliability, and memory usage in the shadow of limited resources of the nodes.

Recently, many researchers proposed compressive sensing paradigm as compression tool, where the base station can reconstruct the transmitted data from

very few samples in case of sparse data, which can be obtained by expansion of the dense data in an orthonormal basis (wavelet, DCT,..., etc). [15, 16].

Compression of data may cause low reliability because the BS may not be able to reconstruct the original data correctly and accurately which known as reconstruction error [17], the probability of the reconstruction error depends on the number of sample, but more samples needs more energy, less samples means less energy consumption but higher reconstruction error probability. To achieve optimal energy efficiency and optimal reliability (which is a function of reconstruction error), we need to find the balance point or set of points that give the user to trade-off between energy efficiency and reconstruction error.

Many research like [18, 19, 20, 21] use different modification of compressive sensing, and adapted the paradigm to fit a specific environment or a specific topology, but they rest on random measurement matrices, others [22, 23] reduce reconstruction error by optimizing the sensing matrix. Most of the research work on side of the balance process, either energy efficiency or reconstruction error.

1.3.2 Objectives

My goal in this research area was to develop a CS-based chain routing algorithm that gives the user the chance to bias towards a specific party of balance equation between energy efficiency and reliability. As-well, The algorithm should constructs the WSN in a way that helps the user to achieve his choices.

The vital contributions of this work are detailed as follows:

- I proposed a multiple objectives genetic algorithm to strike the right balance between the energy efficiency and reconstruction error of the compressive sensing method.
- I used a genetic algorithm to improve the characteristics of the sensing matrix by reducing its mutual coherence.
- I presented a greedy algorithm to split the WSN into multiple chains in a way conserves the balance of the payload, it consists of two sub-algorithms, the first defines the leaf nodes of chains, the second build the chains starting from the leaf nodes.

In addition to WSNs, the developed algorithm can be used as a heuristic optimization algorithm to balance between contradiction factors of the networking process, it can be used in the structure of the graph-based application and optimal resource distribution map.

1.4 Predictive Maintenance systems

1.4.1 Problem statement

Predictive Maintenance (PdM) is mainly concerned with collecting data and estimating the operability of the system under observation. PdMs maximizes the system life cycle, minimizes unplanned downtime, and improves reliability and production quality; it also significantly reduces maintenance costs. To achieve these objectives, it is essential to develop efficient methods to collect data and powerful techniques to predict the operability of the system in the future and estimate the remaining time to failure.

WSNs and IoT [24] provide an effective sensing and large-scale data acquisition system needed by the PdMs to collect the measurements. These measurements are used to describe the working conditions of the maintained systems; they are also used to estimate the future state using the predictive power of neural networks and machine learning models.

Most PdMs use WSNs and IoTs as tools[25, 26]. To the best of our knowledge, there are very few studies interested in finding PdM for WSN, most of them dominating intrusion detection of IoT systems, there is a real need for a PdM system that monitors the operational conditions of the WSNs and estimates the functional status ("OK" or faulty) after some time steps or operational steps, and predict when the probability of faults may exceed predefined threshold.

PdMs of WSNs should be incoherent with the limited resources of WSNs in terms of energy, memory, processing capacity,...,etc. PdMs are curial to help the supervisors of WSN to know when a maintenance action should take place; maintenance actions in WSN may include re-clustering, controlling the movement of mobile base stations, re-deployments of nodes, and many other procedures.

1.4.2 Objectives

My goal in this research was to develop a PdM system based on a prediction model and machine learning algorithm for WSNs; the input features should be the previous K observations of the QoS requirements, include energy efficiency, throughput, reliability, and delay, the input features are formed as a time series. The output should be a vector of length M that represents the operability status of WSN for the next M steps.

The key contributions of this field are summarized as follows:

- I developed a prediction model for PdM systems that estimates the probability that the system remains still fully operational in the next M steps. The system also checks whether the operability in the next M steps is guaranteed with given reliability determined by the parameter ϵ .
- I used the proposed model to develop a PdM system based on a machine learning model for WSNs. It is a streaming real-time data system.

- I adapted the proposed model to be fit for WSNs with limited resources by the design of a proper quantization and encoding scheme, respectively. This can significantly reduce the complexity of the model.
- I implemented the proposed approach by using Feed-Forward Neural Networks (FFNNs).

The predictive model can be used inhumane activities that depends on estimation and prediction, such as stocks and monitoring of environmental and climatic changes, where the proposed PdM model can be deployed in factories or other sites.

1.5 The structure of the dissertation

the remainder of this dissertation is structured as follows, In chapter 2. I briefly introduce WSNs and the routing algorithm that consider the problem of balance between reliability and energy efficiency, then I describe the proposed model of WSNs; I also introduce a mathematical analysis of my proposed algorithm that keeps the residual energy of the path as maximum as possible. Then I present a new routing algorithm that uses Entropy to measure residual energy uniformity. Finally, I compare the performance of the proposed algorithms and other algorithms.

In chapter 3. I define stochastic WSNs, describe queuing model of WSNs, then present a theoretical preface about the principle of backpressure; after that, I describe my proposed algorithm that depends on both backpressure and Entropy of residual energy. Finally, I give numerical results that evaluate the performance of the proposed algorithm.”

Chapter 4 describes the effect of data reduction techniques on reliability and energy efficiency. I present a preface about compressive sensing and MOGA; after that, I introduce my proposed algorithm that uses MOGA to optimize the compressive sensing scheme in terms of reliability and energy efficiency. I describe how to build the WSNs around the proposed algorithm. At last, I evaluate the performance of the proposed algorithm numerically via simulation.

In chapter 5. I present PdMs, their structure, and their advantages, then I give a formal presentation of the prediction model for PdM and implement it by FFNN, I customize the model as a PdM system for WSNs by using quantization and encoding schemes. Then I set up the dataset for FFNN training. Finally, I give the numerical results of a detailed performance of the algorithm under different scenarios.

In chapter 6. I summarize the results, state the thesis, and give a glance at the future.

Chapter 2

Novel reliable and energy-efficient routing protocols for WSN

In this chapter [1], I propose novel energy-aware and reliable routing protocols. The aim is to maximize the lifespan of wireless sensor networks (WSNs) subject to predefined reliability constraints by using multi-hop routing schemes. For the sake of minimizing energy consumption, the source node forwards the packet to the Base Station (BS) via other nodes as relays. In the first proposed protocol, energy efficiency is achieved by maximizing the minimum residual energy of the path subject to fulfilling a predefined reliability constraints. The second protocol is an optimized version of the first one with respect to lifespan and complexity. The optimal path is the one in which the residual energy distribution of the nodes along the path is as close to uniform as possible and the packet arrives at the base station with a given probability of success. To measure the uniformity of the residual energy distribution, I use an entropy-like measure. When implementing the protocol, the information about the current energy state of the network is maintained by using a look-up-table from which the optimal routes are computed on the BS. The BS broadcasts the updated optimal paths to each node after each round of packet transmission.

2.1 Introduction

Packets are sent to the BS either in a single-hop manner or in a multi-hop manner. In the case of single hop, the node sends the generated packet directly to the base station, which require a relatively large amount of energy. While in a multi-hop manner, source nodes send packets to BS via relay nodes forming a multi-node path, where each node in the path forwards the received packet

to another node until it reaches the BS [27]. However, in WSNs packets are transmitted to the BS via unreliable radio channels corrupted by noise and fading [28].

Many techniques have been developed to increase the energy efficiency of WSNs, covering almost all layers of communication protocols ranging from the physical layer to the application layer [29, 30]. Routing is instrumental when designing and operating WSNs; the researchers in [31] describe a group of routing metrics and constraints; they provide a guideline for the composition of these metrics to achieve efficient routing schemes.

Load balancing and reliable information transfer are contributory factors in energy-efficient communications. Lack of load balancing causes the death of some nodes too early, which breaks the connectivity of WSN and reduces the life span of the WSN [32]. The lifespan is defined as the length of time interval until the first node goes flat [33].

The reliability in WSN categorized into three classes: packet reliability, event reliability, and packet-event reliability. In this chapter, I focus on packet reliability. It is classified to up-stream packets, (which mostly represents data packet), and down-stream packets, (which generally represent control packets) [28]. Low reliability leads to high packet loss ratio in the case of low energy consumption. Most protocols achieve energy efficiency by enforcing uniform energy distribution and maximum longevity, but they do not provide reliable links where lost data means wasted energy. Also, they don't use distinct measure for the uniformity of residual energy distribution.

In this chapter, I try to strike a good balance between reliability and energy consumption (which affects network longevity). My concern in the first part of this study is to develop novel routing algorithms that maximize the minimum residual energy subject to a predefined reliability target. To achieve this, I propose the Maximum of the Minimum Residual Energy Protocol (MMREP). According to the proposed method, I select a path by which a packet is sent to the BS over the nodes, of which the minimum remaining energy is the maximum subject to the reliability constraint. The reliability constraint means that the packet will reach the BS with a given probability. This strategy can significantly extend the life span of WSNs.

In the second part of this study, I improve the performance of the basic MMREP in terms of energy efficiency and complexity. I propose an Optimal Residual Energy Based Protocol (OREBP). In OREBP, I use the entropy-like function to measure the uniformity of the residual energy distribution. The packets are sent to the BS in a multi-hop manner. The protocol guarantees that the distribution of the residual energy will remain as close to uniform as possible. Experimental results demonstrated that this method achieves optimal load balancing.

I introduce two novel routing algorithms. They achieve an efficient load balancing under predefined reliability constraint. The key contributions of this chapter are summarized as follows: First, I presented a mathematical analysis for the selection of relay nodes based on the remaining energy and reliability.

Second, I proposed MMREP to select the path that maximizes the remaining energy of its nodes subject to the reliability constraint. Third, I introduced the entropy-like function to measure the uniformity of residual energy distribution. Fourth, I proposed OREBP, which selects the path based on the entropy of residual energy of the path.

The remainder of the chapter is organized as follows:

- In Section 2.2, I provide an overview of related work in the literature.
- In Section 2.3, I introduce the framework in which the different communication protocols will be discussed.
- In Section 2.4, I describe MMREP to select the path which keeps the residual energy as maximum as possible.
- In Section 2.5, I describe OREBP to select the optimal path that maintains uniform residual energy distribution.
- In Section 2.6, I give the numerical results of a detailed performance analysis of the algorithms where their performances are compared with other protocols.
- In Section 2.7, some conclusions are drawn.

2.2 Related Work

Previous works done in the field, classify energy-efficient routing protocols into homogeneous WSN protocols (where all the nodes are identical) and heterogeneous WSN protocols, both types are subdivided into static and mobile [34]. Other researchers classify them based on the network structure (flat and hierarchical); on the communication model (negotiation-based, query-based, and coherent-based); on the topology (location-based and mobile agent-based); or on the reliability (QoS-based and multipath-based) [35].

Some researchers improved the energy efficiency of WSNs by using hierarchical packet forwarding protocols. Low Energy Adaptive Clustering Hierarchy (LEACH) is a hierarchical cluster-based protocol [36, 27]. The network is divided into several clusters; a cluster head (CH) is selected to receive the packets from respective members of that cluster, and then, it forwards the packet directly to the BS. In LEACH, energy efficiency is achieved by clustering and data fusion. By clustering, just the CH has a direct long-distance transmission to the BS; other nodes send their data to CH in short-distance transmission. By data fusion, the CH removes redundant data and fuses the received packets into a single packet. The cluster heads are selected randomly and changed periodically in time to achieve load balancing [36]. However, ignoring the energy level of the cluster head may make it a bottleneck node, and it does not consider reliability.

There are several modified versions of LEACH and these modifications based on three principles, selection of CH, clustering, and data transmission. In LEACH-C, the selection of CH is centric and performed by BS based on the remaining energy and location of nodes [34]. In NEAHC, load balance and energy efficiency are achieved by selecting the CH based on remaining energy. Also, members with low energy are forced to switch between sleep and active status [37]. The authors of [38] propose the selection of CH and clustering based on practical swarm optimization and fuzzy logic; the fitness function depends on the distance between nodes and CH, the distance between CH and BS, and the minimum consumed energy. In [39], the firefly heuristic algorithm is used to select the CH based on distance, energy, and delay. In terms of clustering, CACD algorithm constructs clusters based on the energy depletion of the clusters and node density [40].

Another protocol referred to as PEGASIS is an underlying chain-based routing algorithm [41]. It is a modification of LEACH, the nodes are grouped into a chain using a greedy algorithm, and each of them acts as chain head in turn; each node fuse the received packet into its packets, the fused packet is forwarded to the closest neighbor until it arrives at the chain leader. The chain leader aggregates the packets together and sends them directly to the BS. By selecting the chain leader, regardless of its energy, can result in a shorter lifespan. CHIRON [42] is a chain-based routing protocol that increases energy efficiency by dividing the network field into some smaller areas to create shorter chains for reducing data transmission delay and path duplication. The Energy-Efficient Chain-Based (EECB) routing protocol [43] modified PEGASIS by selecting the chain leader based on the residual energy and distance from the BS. The scheme in [44] selects the chain leaders from a group of nodes in the neighborhood of BS. The authors of [45] used a heuristic Ant Colony optimization to reduce delay and to find the optimal path with minimum transmission distance.

As far as the reliability is concerned in [33], an energy-efficient and reliable protocol is proposed achieving a predefined reliability by multipath routing. But, only two-hop paths are assumed. Researchers of [9], improve the reliability by finding the best location of CH. RE-AEDG [8] and GIN [10] protocols use the cooperative routing model to enhance reliability, where the transmitted packet is overheard by several neighbors and retransmitted by several neighbors too. Some researches use network coding to improve reliability [46, 18]. Most of these protocols improve reliability by increasing the redundant data, which means wasted energy.

Unfortunately, these protocols do not achieve optimal energy balancing and do not take reliability into consideration when selecting an energy-efficient path. As a result, further investigation is needed into developing packet forwarding mechanisms.

2.3 The model

For developing new routing algorithms I consider WSNs as a graph $G(V, E, d)$, where set V refers to the nodes $V = |N|$, while set E denotes the edges and $d_{ij}, (i, j = 1, \dots, N)$ are the distances between the nodes, which are arranged in a distance matrix, distance represents edge weight. packets are forwarded from the nodes to the BS in the form of multihop routing. In multihop protocols, the nodes use each other to relay the packets until they finally are received by the BS.

Based on the Rayleigh fading model [43], the energy g_{ij} needed for transmitting a single packet from node i to node j over distance d_{ij} with the probability of successful transmission P_{ij} is given as

$$g_{ij} = -d_{ij}^\alpha \frac{\theta \sigma_Z^2}{\ln(P_{ij})} \quad (2.1)$$

where θ is the so-called modulation and coding constant [47], σ_Z^2 denotes the power of noise, α is the large-scale path loss exponent (usually $2 \leq \alpha \leq 6$). The reliability constraint is expressed by a predefined data loss percentage ε ; where the probability of overall success of packet transmission from the source node to the BS is supposed to fulfil $\prod_{j=0}^m P_{l_j l_{j+1}} = 1 - \varepsilon$, in case of using m relay nodes to forward packets to the BS. I adjust g_{ij} to get same P_{ij} for all nodes, $P_{ij} = \sqrt[m]{1 - \varepsilon}$.

The energy state of the network at time instant k is described by an energy state vector $\mathbf{c}(k) = (c_1(k), \dots, c_N(k))$ where $c_i(k)$ represents the available battery power at node i at time instant k . I assume that all the nodes have the same initial energy at time instant 0, i.e. $c_i(0) = E \forall i = 1, \dots, N$.

When seeking a path of maximum m relay node from the source node s to the BS, we search for a set of indices $R = \{l_1, \dots, l_m\}$ referring to the relay nodes participating in the packet transfer, where the source node s sends the packet to node l_1 then the packet is forwarded to node l_2 and so on, and finally from node l_m to the BS. Each node has a copy of the routing table which is calculated and broadcasted by BS.

In the forthcoming discussion, I assume that the BS has full information on the distance matrix of the network and, as a result, it has the routing table of all nodes. The BS updates its version of the energy state vector after receiving each packet, by which it perceives of the remaining energy and the existence of the nodes. With each round, it runs the proposed algorithms to calculate the optimal path from each node according to the energy state vector, the predefined data loss percentage (ε), and predefined maximum number of hops. The updated routing table is then broadcast to all nodes. I also suppose deterministic transmission, where the nodes transmit the same amount of data in a predefined frequency (periodically), and each node forwards packets according to its routing table.

One may say that this procedure will increase the energy consumption due

to the frequency of BS broadcasting. However, the energy needed for receiving is significantly less than the energy needed for transmission [40]; it does not depend on the distance; it just depends on the size of the received information. The information received by the nodes is a short control message; it consists of the ID of the node, the ID of the next relay node, and the ID of the source node. Moreover, nodes eliminate the overhead required for chain or path setup, as with PEGASIS and LEACH [33, 40]. The BS, with its high energy and high computational capabilities, does all the work—this configuration is used for the majority of IoT systems. Besides that, the proposed algorithms reduce the consumed energy in overhearing reception; the node receives just from the relay node specified by BS deterministically, which makes the algorithms beacon-less because the nodes don't need to exchange routing information.

In the proposed protocols, aside from the necessity of determining the optimal energy consumption values that maximizes the residual energy of the path, it is also necessary to guarantee that the packets arrive at the BS with a given reliability $(1 - \varepsilon)$.

2.4 Routing to Maximiz the Minimal Residual Energy

In this section, I select a path in which a packet is sent to the BS with the minimum remaining energy being maximized subject to the constraint that the packet will reach the BS with a given probability.

Thus, the path over which the packet is forwarded to the BS (denoted by \mathfrak{R}) is optimal if

$$\mathfrak{R} : \max_{\mathfrak{R}} \min_i c_i(k+1) \quad (2.2)$$

subject to the constraint $\prod_{j=0}^m P_{l_j l_{j+1}} = 1 - \varepsilon$. This strategy can easily be implemented when the hop count of packet transfer has been set prior to sending the packet to the BS.

2.4.1 2-hop routing

Let us assume that packets are forwarded to BS in a two-hop path, and the sender node is denoted by s , while the intermediate node relaying this packet to the BS is node l . Then there are two components that change in vector $c(k+1)$ compared to $c(k)$,

$$c_s(k+1) = c_s(k) - \left(-d_{s,l} \frac{\Theta \sigma_Z^2}{\ln(P_{sl})} \right). \quad (2.3)$$

and

$$c_l(k+1) = c_l(k) - \left(-d_{l,BS} \frac{\Theta \sigma_Z^2}{\ln(P_{l,BS})} \right). \quad (2.4)$$

Furthermore, if one wants to ensure that the packet sent by node s reaches the BS with a given probability $1 - \varepsilon$, then the reliability constraint can be expressed $P_{s,l} * P_{l,BS} = 1 - \varepsilon$.

Thus, in the case of two-hop routing, one has to determine the relay node l , by solving the following constrained optimization problem:

$$l_{opt} : \max_l \min \left\{ c_s(k) - d_{sl} \frac{\Theta \sigma_Z^2}{\ln(P_{sl})}, c_l(k) - d_{l,BS} \frac{\Theta \sigma_Z^2}{\ln(P_{l,BS})} \right\}. \quad (2.5)$$

subject to the constraint $\ln(P_{s,l}) + \ln(P_{l,BS}) = \ln(1 - \varepsilon)$.

Solution

Let us assume that we have chosen a relay node denoted by index l , then the condition $P_{sl} * P_{l,BS} = 1 - \varepsilon$ can be paraphrased as:

$$\ln(P_{s,l}) + \ln(P_{l,BS}) = \ln(1 - \varepsilon). \quad (2.6)$$

From Eq. (2.1)

$$-\frac{d_{s,l}^\alpha}{g_{s,l}} \theta \sigma_Z^2 - \frac{d_{l,BS}^\alpha}{g_{l,BS}} \theta \sigma_Z^2 = \ln(1 - \varepsilon). \quad (2.7)$$

From Eq. (2.7), we can define $\varphi(g_{sl})$ which expresses the relationship between $g_{l,BS}$ and $g_{s,l}$ due to the constraint as:

$$\varphi(g_{sl}) = g_{l,BS} = \frac{-g_{sl} d_{l,BS}^\alpha \theta \sigma_Z^2}{g_{sl} \ln(1 - \varepsilon) + d_{l,BS}^\alpha \theta \sigma_Z^2}. \quad (2.8)$$

The minimum residual energy (if node l is selected) will be maximum if the residual energy at the source node and the relay node are equal to each other $c_s(k+1) = c_l(k+1)$. So from Eq. (2.3) and Eq. (2.4): $c_s(k) - g_{s,l} = c_l(k) - g_{l,BS}$ and by using Eq. (2.7) and Eq. (2.8), we have:

$$c_s(k) - g_{s,l} = c_l(k) - \varphi(g_{sl}). \quad (2.9)$$

One can use Eq. (2.8) to determine $g_{l,s,opt} : c_s(k) - g_{s,l} = c_l(k) - \varphi(g_{sl})$. since the source node s is given, one can calculate $g_{l,s,opt}$ for all possible relay nodes $l = 1, \dots, N$. Then l_{opt} can be selected by solving the problem

$$l_{opt} : \max_{l \in \{1, \dots, N, l \neq s\}} c_s(k) - g_{sl,opt}. \quad (2.10)$$

This requires the solution of Eq. (2.9) $N - 1$ times and then $N - 1$ comparison of the value $c_s(k) - g_{l,s,opt} \forall l = 1, \dots, N, l \neq s$.

2.4.2 Extension to m-hop routing

Based on this reasoning, the protocol can easily be extended to m-hop routing, as well. Now, I assume that packets are forwarded to BS via an m-hop path and the sender node is node $l_0 = s$ and the sink node is the BS denoted by $BS = m + 1$. The path containing the sender and the relay nodes is denoted by $\mathfrak{R} = \{l_0, \dots, l_m\}$.

Then there are $m+1$ components changing in vector $\mathbf{c}(k+1)$ compared to $\mathbf{c}(k)$, given as $\mathbf{c}_l(k+1) = c_l(k) |g_{l_j, l_{j+1}}, j = 0, \dots, m$. Similarly to the previous constraint, if one wants to ensure that the packet sent by node s reaches the BS with a given probability $(1 - \varepsilon)$, then the reliability constraint can be expressed as $\prod_{j=0}^m P_{l_j l_{j+1}} = 1 - \varepsilon$. Thus, in the case of m -hop routing, one has to solve the following constrained optimization problem:

$$l_{1opt}, l_{2opt}, \dots, l_{m-opt} : \max_{l_2, l_3, \dots, l_m} \min_{j \in \{0, \dots, L\}} \left(c_{l_j}(k) - d_{l_j, l_{j+1}} \frac{\Theta \sigma_Z^2}{\ln(P_{l_j, l_{j+1}})} \right). \quad (2.11)$$

subject to the constraint $\prod_{j=0}^m P_{l_j l_{j+1}} = 1 - \varepsilon$.

Solution

let us assume that one has chosen a path $\mathfrak{R} = \{l_0, \dots, l_m\}$. Along this path the reliability constraint $\prod_{j=0}^m P_{l_j l_{j+1}} = 1 - \varepsilon$ can alternatively be given as $\sum_{j=0}^m \ln(P_{l_j l_{j+1}}) = \ln(1 - \varepsilon)$.

From Eq. (2.1) it becomes:

$$\sum_{j=0}^m \ln(P_{l_j l_{j+1}}) = - \sum_{j=0}^m \frac{d_{sl_j}^\alpha}{g_{sl_j}} \theta \sigma_Z^2 = \ln(1 - \varepsilon). \quad (2.12)$$

The minimum residual energy will be maximum if the residual energy at all the relay nodes are equal to each other $\mathbf{c}_{l_j}(k+1) = c_{l_{j+1}}(k+1), j = 0, \dots, m$. So from Eq. (2.3) and Eq. (2.4):

$$c_{l_j}(k) - g_{l_j, l_{j+1}} = c_{l_{j+1}}(k) - g_{l_{j+1}, l_{j+2}}. \quad (2.13)$$

The $m+1$ equations above yield a solution to $g_{opt} = (g_{l_0 l_1, opt}, \dots, g_{l_m BS, opt})$. One can calculate g_{opt} for all possible m hops paths fixing the path $\mathfrak{R} = \{l_0, \dots, l_L\}$, where L is the actual number of hops. Then the optimal path $\mathfrak{R}_{opt} = \{l_0, l_{1, opt}, \dots, l_{m, opt}\}$ can be sought by solving the following problem:

$$\mathfrak{R}_{opt} = \{l_0, l_{1, opt}, \dots, l_{m, opt}\} : \max_{\mathfrak{R}} (c_{l_0}(k) - g_{l_0 l_1, opt}). \quad (2.14)$$

This requires the solution of Eq. (2.12) $\sum_{L=1}^m \frac{(N-1)!}{L!(N-1-L)!}$ times and then

$\sum_1^m \frac{(N-1)!}{L!(N-1-L)!}$ comparisons of the value $c_{i_0}(k) - g_{l_0 l_1, opt}$. I refer to this method of routing the packets to the BS as the Maximum of minimum Residual Energy Protocol (MMREP). Figure 2.1 shows the steps of MMREP summarized in a flowchart:

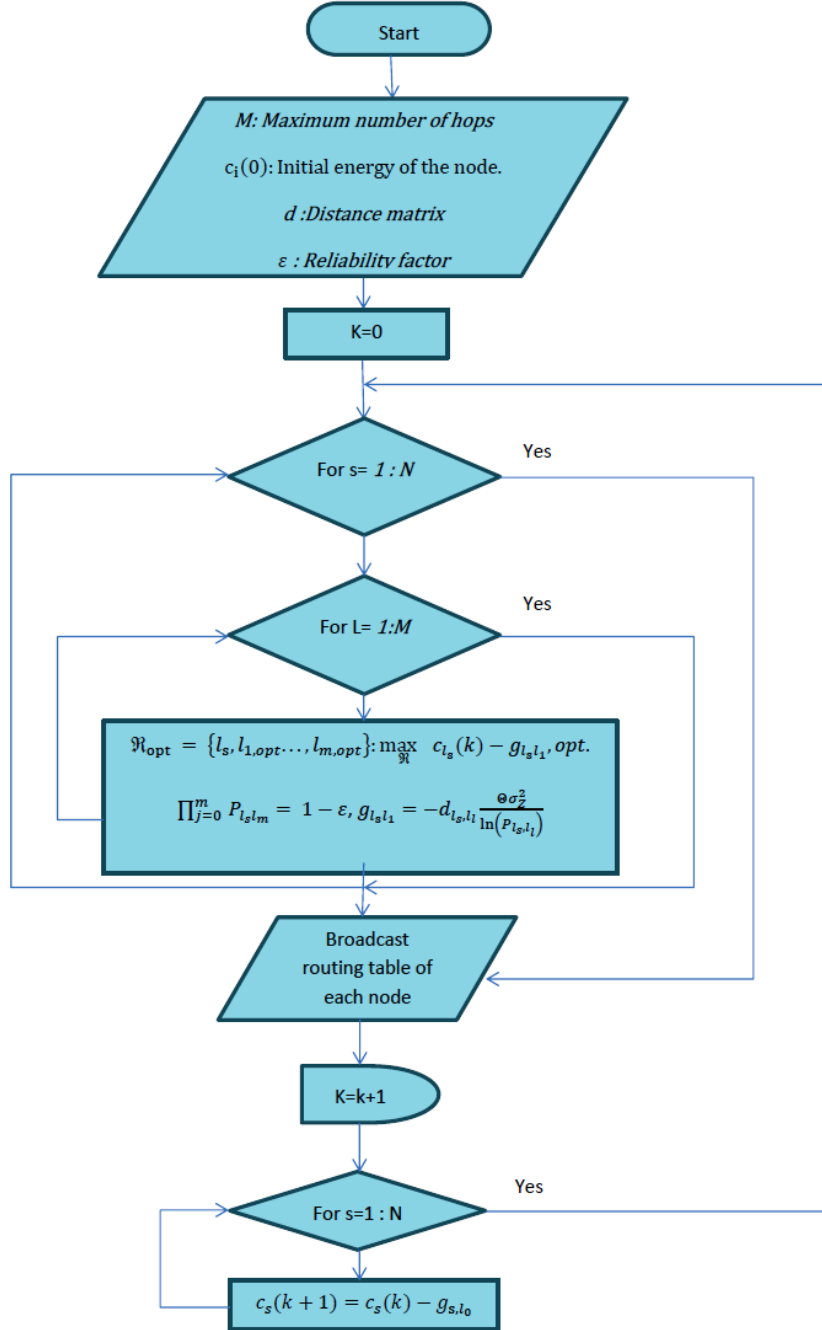


Figure 2.1: The flowchart of MMREP

2.5 Optimal path selection based on networks state energy entropy

We can further generalize this method by introducing an entropy-like measure on energy distribution. The equality of residual energy of relay nodes can be extended for uniform distribution of residual energy across all the nodes of the path. This uniformity can be achieved by maximizing the entropy of the residual energy. At instant ($k = 0$)), the entropy is maximum because all the nodes have the same initial energy; to keep the entropy at maximum, the gradient of the entropy should be minimized. We can calculate the entropy of the normalized residual energy distribution on the nodes of the network $\bar{c}(k) = c_1(k), c_2(k), \dots, c_N(k)$ at time instant k . Hence, the entropy characterizing the current energy distribution is:

$$H(\bar{c}) = \sum_{i=1}^N \frac{c_i(k)}{\sum_{j=1}^N c_j(k)} \log \frac{\sum_{j=1}^N c_j(k)}{c_i(k)}. \quad (2.15)$$

The gradient of the entropy can be calculated as follows

$$\bar{V} = \nabla H(\bar{c}) = \frac{\partial H}{\partial c_1}, \frac{\partial H}{\partial c_2}, \dots, \frac{\partial H}{\partial c_N} \quad (2.16)$$

Suppose that source node s is sending a packet via the relay nodes l_1, \dots, l_m where l_m is the last relay node, then the residual energy of nodes involved in packet forwarding will change as:

$$c_{l_j}(k+1) = c_{l_j}(k) - \left(-d_{l_{j+1}} \frac{\Theta \sigma_Z^2}{\ln(P_{l_{j+1}})} \right). \quad (2.17)$$

Further, the gradient of the entropy of the energy distribution on relay node is given as

$$\frac{\partial H}{\partial c_l} = \frac{\sum_{j=1}^N c_j(k) - c_l(k)}{\left(\sum_{j=1}^N c_j(k)\right)^2} \log \left[\frac{\sum_{j=1}^N c_j(k)}{c_l(k)} \right] + \frac{c_l(k) - \sum_{j=1}^N c_j(k)}{\left(\sum_{j=1}^N c_j(k)\right)^2}. \quad (2.18)$$

while the gradient of the entropy of the energy distribution on the other nodes of the path is given as

$$\frac{\partial H}{\partial c_i} = \frac{-c_i(k)}{\left(\sum_{j=1}^N c_j(k)\right)^2} * \text{Log} \left[\frac{\sum_{j=1}^N c_j(k)}{c_i(k)} \right] + \frac{c_i(k)}{\left(\sum_{j=1}^N c_j(k)\right)^2}. \quad (2.19)$$

Thus, the change of the entropy is:

$$\Delta \mathcal{H} = \bar{V}^T * \Delta \bar{c}(k).$$

The Optimal path is the one that keeps this gradient minimum, thus enforcing only small changes in the energy entropy, guaranteeing that the energy distribution falls as close to its maximum as possible. In this way, a more or less uniform residual energy distribution can be maintained, while sending packets to the BS. The optimum choice of relay nodes (the optimal path) which minimizes the change in the gradient can be obtained as:

$$l_{opt} = \min(\Delta H = \bar{\mathbf{V}}^T * \Delta \bar{c}(k)). \quad (2.20)$$

After each round the BS updates the residual energy vector of all nodes, $\mathbf{c}(k) = c_1(k), \dots, c_i(k), \dots, c_N(k)$, and find the optimal path from node i to BS, and calculate the relay nodes of the path using OREBP. The BS broadcasts a new look-up-table containing the optimal routes of each node. This can help reduce the computational load on the nodes and improve their energy efficiency.

The needed calculations increase as the number of maximum hops (m) increases; as the maximum number of hops (m) increases, more possible paths are available and more optimality is achieved.

I refer to this method of routing the packets to the BS as the Optimal Remaining Energy-Based Protocol (OREBP), The steps of the algorithm are summarized by the flowchart shown in Fig 2.2.

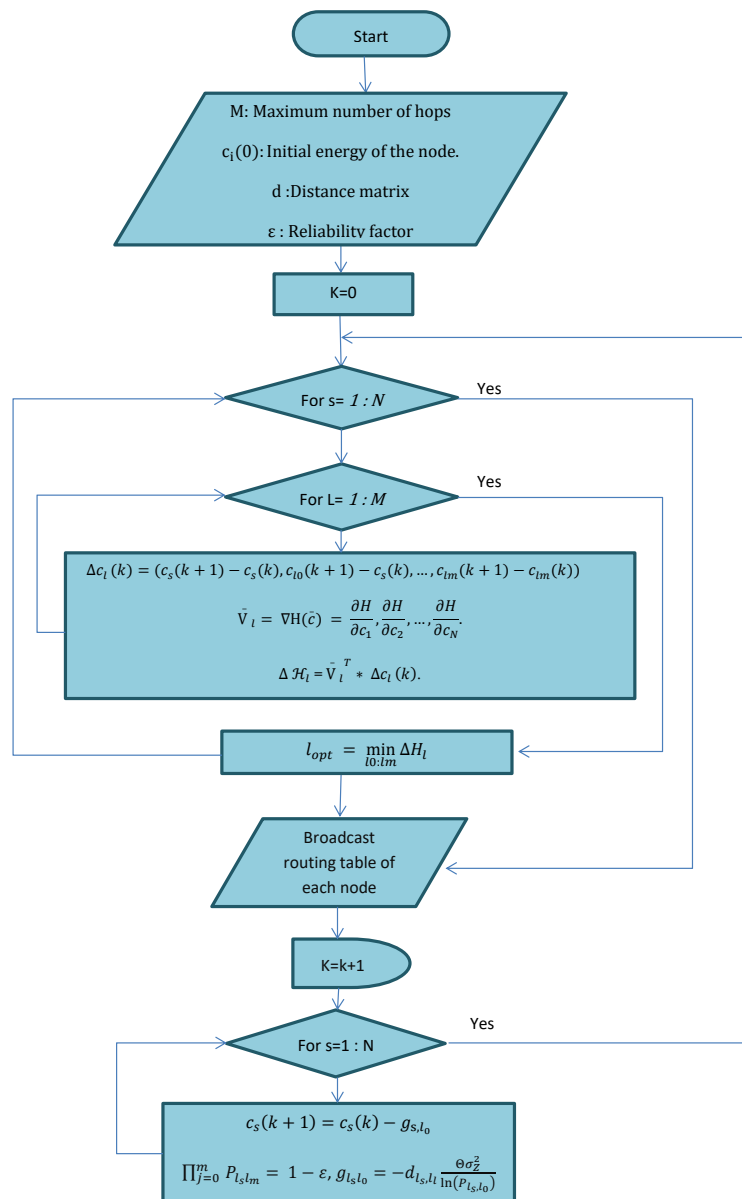


Figure 2.2: The flowchart of OREBP

2.6 Simulation results

I used MatLab R2018b installed on a personal laptop to evaluate the performance of the proposed algorithms MMREP and OREBP, respectively in comparison to the PEGASIS protocol. In PEGASIS, each node forwards the packet to its closest neighbor; when the packet arrives at the chain leader, it sends the packet directly to the base station.

PEGASIS does not take reliability into consideration, whereas MMREP and OREBP do. To introduce reliability constraints to PEGASIS and thus make it comparable with the proposed protocols, it is modified as follows: each packet should be forwarded to the BS with at least success probability $P_{ij} = (1 - \varepsilon)$, where i is the source node and j is the destination node. From Eq. (2.1); $P_{ij} = -d_{ij}^\alpha \frac{\theta \sigma_Z^2}{g_{ij}}$ is calculated as in [47]. In case of an m -hop path, $P_{path} = \sqrt[m]{1 - \varepsilon}$. For the whole round, I calculated the average success probability of all paths. Figure 2.3 shows how 100 nodes are deployed randomly onto a grid of $100 \text{ m} \times 100 \text{ m}$ according to a 2D normal distribution; the BS is allocated in the upper-right corner of the sensing field, as shown by the green point. while Fig.2.4 shows the network as a graph. The highlighted chain is constructed by the Dijkstra shortest path algorithm, as defined by the base station.

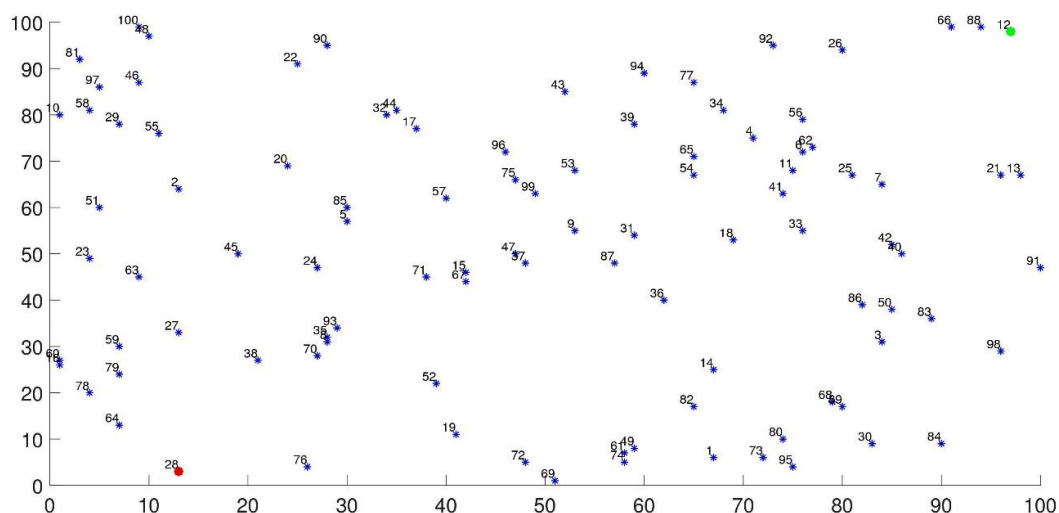


Figure 2.3: Simulated Network

The transmission energy will be calculated according to Rayleigh fading model Eq. (2.1) under the following assumptions:

- Neglecting Conditioning energy needed by the electronics.
- $\Theta = 10^{-6}$.
- $\sigma_Z^2 = 0.1$.
- $\alpha = 2$.
- P_{ij} is calculated based on reliability $(1 - \varepsilon)$ and the number of relay nodes m , where $\prod_{j=0}^m P_{l_j l_{j+1}} \geq 1 - \varepsilon$. I assumed that P_{ij} is the same for all nodes, then $P_{ij} = \sqrt[m]{1 - \varepsilon}$.

- All the nodes have the same initial energy.
- The lifespan is the number of packets transmitted over the chain until the first node goes flat. Each transmission round carries a single packet.
- Each node has an adjustable power transmission, which enables it to access any node in the chain including the BS..

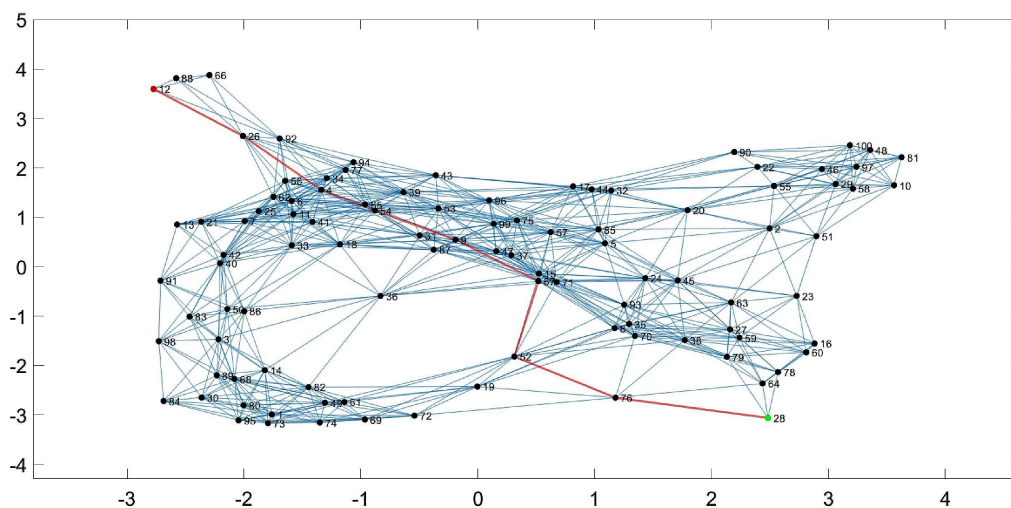


Figure 2.4: The network as a graph

The path shown in Fig. 2.4 consists of seven nodes, numbered from 1 to 7, where node number 1 is the farthest from the BS, it is shown as red point in Fig. 2.3. Figure 2.5 shows the residual energy of the nodes in the chain for MMREP, OREBP, and PEGASIS respectively, when the first node goes flat. In the case of MMREP and OREBP, one can note that almost all the nodes have the same level of residual energy, the residual energy is distributed more uniformly (near zero) for all nodes, In the case of PEGASIS, one can note also note that as the node gets further away from the BS, it loses energy at a higher rate, the residual energy of the nodes decreases as the nodes gets farther from the BS. So, one can note unevenly distribution of the residual energy.

Figure 2.6 shows the life span of a network which uses MMREP, OREBP, PEGASIS, subject to a reliability constraint, with different data loss percentage ε values, 0.025, 0.05, 0.075, 0.1, 0.125 and 0.15, respectively.

Since PEGASIS does not consider ε when calculating transmission energy, the figure shows that it has a higher life span than MMREP and OREBP; it uses lower transmission energy, which causes a higher packet loss percentage, which means lower reliability; hence, the life span of a network with PEGASIS is the same regardless of data loss percentage ε . In the case of the proposed algorithms, however, life span increases with increasing data loss percentage ε , while the probability of successful packet transmission $(1 - \varepsilon)$ is decreasing.

With MMREP and OREBP, the life span is a function of ε . The users can trade-off between the reliability represented by ε and the life span.

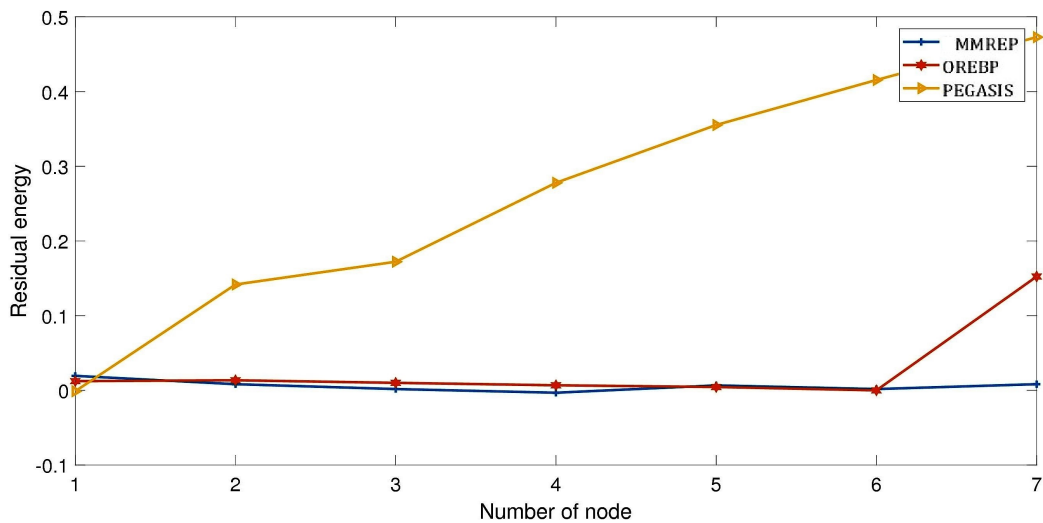


Figure 2.5: Distribution of Residual Energy of the path at the moment when the first node goes flat.

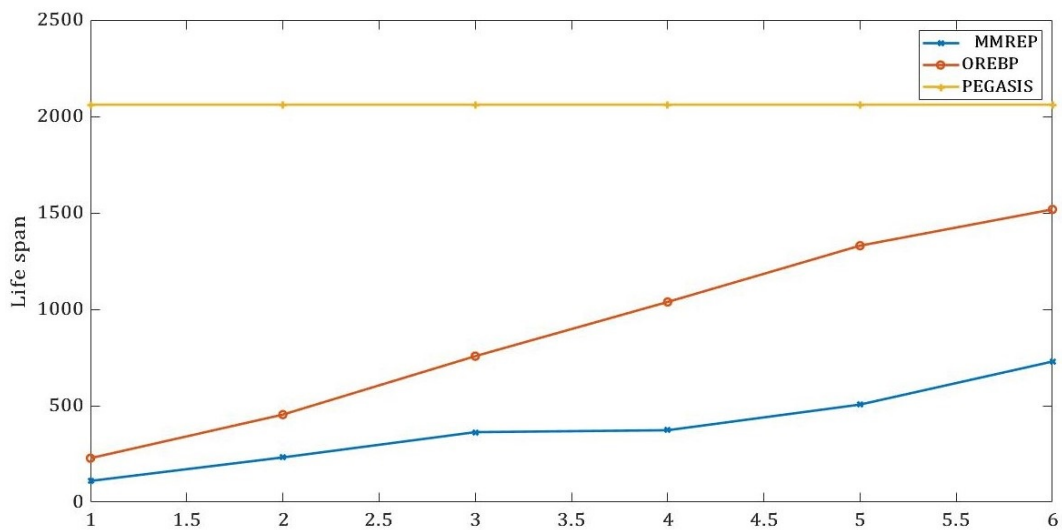


Figure 2.6: Lifespan with different data loss (ε).

Figure 2.7 shows the success probability with OREBP and PEGASIS. I assumed a predefined data loss rate ($\varepsilon = 0.05$) for OREBP. The success probability for PEGASIS is calculated as mentioned above. The figure shows the success probability against a variable number of hops ranging from one to six. OREBP always has a higher success probability than PEGASIS, although the value depends on the number of hops. The figure shows that using OREBP improves the reliability from 63% in case of a one-hop path to 71% in case of a

six-hop path. Lost data means wasted energy. The figure shows that the success probability decreases with an increasing number of hops. In OREBP, the operator can control the success probability by trading off between predefined data loss ε with life span, as shown in Fig. 2.6.

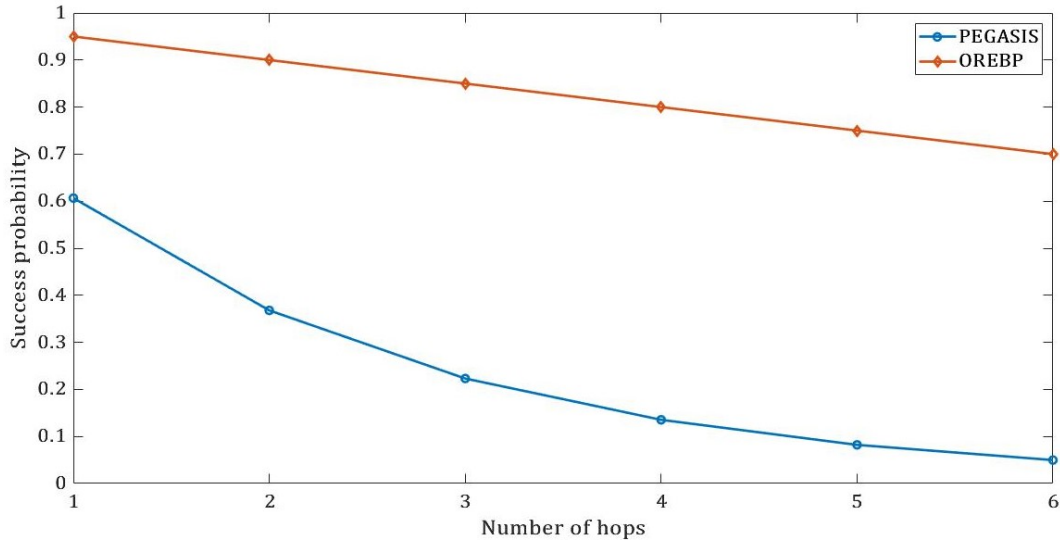


Figure 2.7: Success probability when data loss ($\varepsilon = 0.05$).

Figure 2.8 shows the relationship between the life span and the number of hops in the case of MMREP, OREBP, and PEGASIS, respectively. I assumed a predefined data loss rate ($\varepsilon = 0.05$). In the case of MMREP, OREBP, the consumed energy depends on ε . The probability of the overall success of packet transmission from the source node to the BS should satisfy $\prod_{j=0}^m P_{l_j l_{j+1}} = 1 - \varepsilon$, where m is the number of the hops.

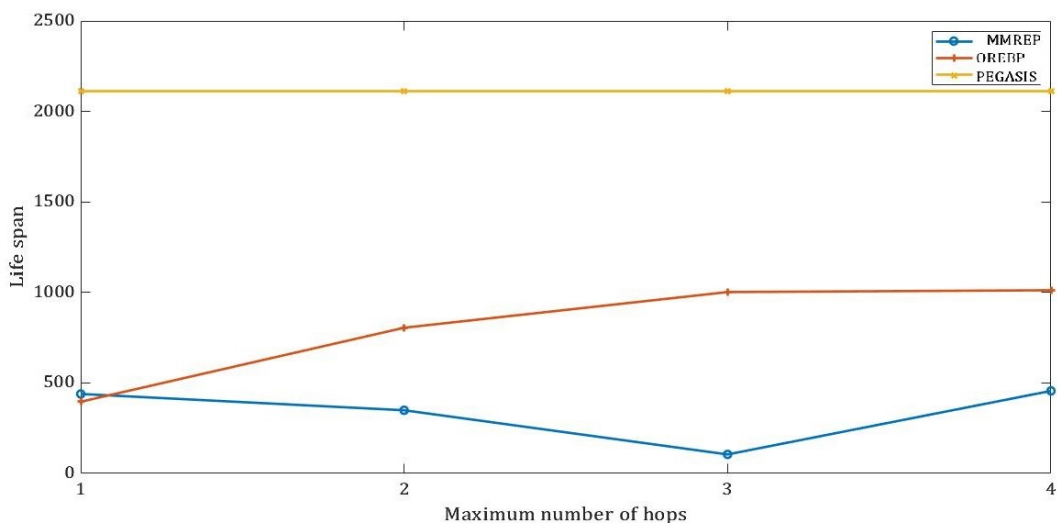


Figure 2.8: lifespan & maximum number of hops ($\varepsilon = 0.05$).

The figure shows the life span against a variable number of hops ranging from one to four. We note that the life span with OREBP increases as the number of hops increases. But, in the case of MMREP, the relationship is unstable because the set of Eq. (2.12) is solved numerically, where a set of solutions are found, so, one needs an algorithm to pick the suitable solution. In OREBP, life span increases as the number of maximum hops increases because there are more paths available to select from. With PEGASIS, the consumed energy is a function of single transmissions rather than the path overall transmissions, so it does not depend on the number of hops.

As the number of hops increases, not only the life span becomes longer, but also the complexity of the algorithm increases. Figure 2.9 shows the relationship among the maximum number of hops m , on the x-axis, the life span y-axis on the left, and the execution time (a proxy for the complexity) on the secondary y-axis on the right. As expected, the figure shows that in OREBP as the number of hops increases, the life span increases, but the complexity of the protocol also increases. The user has the choice of trade-offs between the life span, the complexity, and the maximum number of hops.

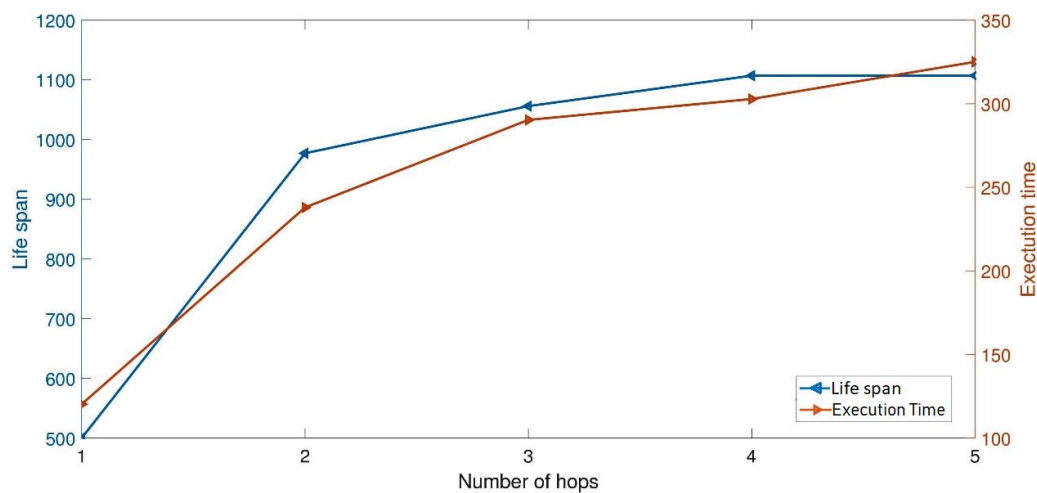


Figure 2.9: Lifespan, Number of hops and Execution time.

All the figures show higher a life span with PEGASIS, but also a high percentage of lost data, which implies wasted energy. The results show that the performance of OREBP is better than the performance of MMREP. As mentioned, MMREP is the basic idea of the algorithm, but I used the entropy principle to optimize it and to reduce its complexity.

2.7 Conclusions

In this chapter, I introduced a protocol which maximizes the minimum residual energy subject to a predefined reliability constraint. I optimized the energy balancing of WSNs by a new protocol (OREBP) to achieve uniform

energy consumption by maximizing an entropy-like measure. The performance of the proposed protocols was compared to the performance of PEGASIS. As demonstrated by the simulation results, the proposed protocols improve the energy efficiency of WSNs by maximizing the life span, subject to a reliability constraint. They also improve the energy consumption distribution among the nodes of the network. As shown, the proposed protocols still have a high degree of complexity. I defined four performance parameters: life span, data loss percentage, the maximum number of hops, and the complexity. The network operator has the choice to control the performance by striking an optimal trade-off among these parameters. To achieve this required the solution of a constrained optimization problem.

Chapter 3

A new energy-efficient and reliable protocol for stochastic WSNs

In this chapter [2], I take into account that the packets are generated, buffered, and transmitted subject to queuing dynamics; thus, I investigate the stochastic WSNs. A new energy-efficient and reliable routing protocol is introduced by using a stochastic traffic generation model and a wake-up/sleep (ON/OFF) mechanism. As before, my objective is to improve the longevity of the WSNs by energy balancing and providing reliable packet transfer to the base station at the same time. The proposed protocol is based on the principle of the back-pressure method. In order to optimize energy consumption, I use a cost function related to entropy like function defined over the residual energies of the nodes. In the case of two-hop routing the optimal relay node is selected as the one which has maximum backlog difference and keeps the distribution of residual energy as close to uniform as possible. The uniformity is measured by the change of the entropy-like function defined over the residual energies of the nodes. This protocol also assumes the Rayleigh fading model, it guarantees that the transmission energy is enough to ensure arriving the packet at the base station with a predefined probability of success. Simulation results show that the proposed algorithm significantly improves the performance of traditional back-pressure protocol with respect to energy efficiency, E2E delay and throughput, respectively.

3.1 Introduction

While in deterministic WSNs the nodes transmit the data according to a predefined schedule, in the case of stochastic WSNs the nodes randomly

transmit data subject to queuing dynamics; the MAC manages to build in queues-like where the data is held before transmission. Some WSNs applications require this paradigm such as surveillance and event-based applications.

Queueing theory is widely used in the design and analysis of WSNs [48]. In [49] queueing models are used to characterize End-to-end delays in WSNs. In [50] it is also used to detect and control the congestion of WSNs. Furthermore, queueing theory is used in [51] to improve and analyze the energy efficiency of WSNs by controlling ON/OFF schemes. Paper [52] uses queueing theory to improve the reliability by adjusting the arrival rate and service rate with queue length to prevent congestion which causes packet loss because of overflowing queues.

There are many routing protocols based on the Back-Pressure algorithm [53], where the WSNs are considered as a network of queues. The routing and forwarding decisions are made independently for each packet by computing the back-pressure weight for each outgoing link. The back-pressure weight is a function of the difference between the backlogs of both ends of the link, estimated link rate, and Link cost (Penalty) function, respectively.

In this chapter, I propose Back-Pressure and Energy Entropy Based protocol (BPEEBP), a new energy-efficient and reliable routing algorithm which is based on back-pressure and the entropy of the residual energy. Besides the difference of the backlog of both ends of the link, BPEEBP uses a cost function depending on the change of entropy of the residual energy of the nodes, the higher this entropy is, the closer we get to uniform energy distribution among the nodes.

I assume that the nodes generate packets randomly, subject to a stochastic model, and their activities are controlled by an ON/OFF scheme. As far as the radio propagation is concerned, I use the Rayleigh fading model. BPEEBP aims to increase longevity and guaranteeing a given level of reliability at the same time. The proposed protocol improves energy efficiency concerning the total consumed energy and enforcing uniform energy distribution among the nodes; thus eliminating the formation of the bottleneck nodes. Also, it is aware of influential issues such as the limited storage space of the node, end-to-end delay, and congestion occurrence.

The remainder of this chapter is divided into six sections:

- In section 3.2, I give an overview of the related work.
- In section 3.3, I introduce my model for the WSN.
- In section 3.4, I present a description of the routing protocol.
- In section 3.5, I give the numerical results of a detailed performance analysis of the algorithms where their performances are compared with other protocols.
- In Section 3.6, I draw some conclusions.

3.2 Related work

The Back Pressure protocol developed to achieve energy efficiency and taking the data traffic into account was first proposed by L. Tassiulas and A. Ephremides in 1990. Some Back-pressure based protocols used in WSNs are centralized, where the routing and scheduling decisions are taken by the Base Station [54]. The shortest-path-aided back-pressure algorithm (SBA) introduced in [55] is a centralized protocol and it aims to reduce end-to-end (E2E) delay, not only the backlog difference and estimated link rate are taken into consideration, but also the number of hops from the source node to the destination.

Back-pressure Based Collection Protocol (BCP) is a distributed back-pressure based protocol [11], routing, and scheduling decisions are taken by the node itself, besides backlog, it uses the expected number of transmission (EXT) as a penalty (cost) function. BCP uses LIFO (last in, first out) queue structure to reduce E2E delay, it uses the floating-queue idea to solve the problem of the packets arrived early, these packets may be trapped at some relay nodes, by floating queues, they are discarded and moved to a virtual queue [11].

In traditional back-pressure algorithms, a small backlog difference may cause a selection of long paths but Packet-by-Packet Adaptive Routing and scheduling algorithm (PARN) addresses this issue [12]. An M-back-pressure mechanism is introduced where the link is scheduled to be active only if the difference between the length of the queues of the source node and the destination node is larger than M , ($M > 1$). But, this may increase E2E delay in case of light and moderate traffic. To solve this problem and to reduce queues complexity (i.e. the number of queues in each node), adaptive routing is proposed, where each node has an actual queue for each neighbor, and shadow queues for each node in the networks, shadow queues are just counters. The back-pressure of the links depends on the difference of the counters; packets are served from the real queue at the link in a first-in, first-out method (FIFO).

The protocols mentioned above aim at improving the E2E delay and reducing the complexity, but gradient assisted energy-efficient back-pressure scheduling algorithm (GRAPE) seeks to improve the energy efficiency of WSNs [13]. In GRAPE, the weight of the link is determined on the basis of the differential backlogs for the source and the destination, the residual energy status of the destination, and their gradient difference, where the gradient is the hop count between the node and the base station, the nodes forward the packet to the neighbor which has higher residual energy and closer to the base station.

Multi-factors back-pressure scheduling algorithm (MFBS) presented in [14] uses the same idea, but it takes the distance between nodes into the account instead of the number of hops. But residual energy and the distance from the base station do not guarantee the absence of a bottleneck node, because they do not ensure a uniform distribution of residual energy.

Other research improve the performance of back-pressure by combining it

with other techniques. NCBPR [56] combines bac-kpressure with network coding; it uses network coding to reduce congestion, redundant data, where a back-pressure algorithm is used for routing scheduling to guarantee load balancing among the nodes. BRPL [57] conglomerates back-pressure algorithm and RPL (routing protocol for low-power and lossy networks), it switches between them based on the status of the network, routing decisions are taken depending on both gradients of the backlogs (related to back pressure); switching between them solve the problem of the poor performance of RPL in terms of network dynamics and throughput. Such algorithms are used in large scale IoT, they are complicated and difficult to be implemented in traditional WSNs.

In this study, I combine the back-pressure protocol with energy balancing to prolong the life span of WSNs; energy balancing will be controlled by entropy-like measure. Reliability is maintained by assuming the Rayleigh fading model; it determines the transmission energy needed to guarantee that the packet arrives at the base station with a predefined probability of success.

3.3 The system model

In this chapter, I suppose the model of WSNs described in section 2.3. My algorithm is a chain based algorithm; the base station constructs a set of chains cover all the nodes of the WSN. The chains are constructed by the Dijkstra shortest path algorithm, and the BS has full vision of distance matrix between nodes, the distance between two nodes represents the weight of the corresponding edge in the graph of the network. The path from the source node i to the BS is described as a set of m relay nodes participating in the packet transfer, $\mathfrak{R}_{i-BS} = \{i, r_1, \dots, r_m, BS\}$.

Based on the Rayleigh fading model [43], the energy g_{ij} needed for transmitting a single packet from node i to node j with the probability of successful transmission P_{ij} is given by Eq.(2.1), in a two-hop path, there are two components that change in energy state vector $c(k+1)$ compared to $c(k)$ as shown in Eq.(2.3) and Eq.(2.4). To ensure that the packet sent by node i reaches node j with a given probability $1 - \varepsilon$, then the reliability constraint can be expressed $P_{i,j} = 1 - \varepsilon$.

In the case of stochastic WSN, the network can be considered as an open network of $M / M / 1$ queues; each queue has two types of arrivals:

1. External arrivals follow Poisson process at a rate of γ_i ; they represent the generated packets by the node, the vector $\boldsymbol{\gamma}$ represents the generation rate of the chain.
2. Internal arrivals represent the received packets from other, where λ_i is the total arrival rate of both external and external arrivals, the vector $\boldsymbol{\lambda}$ represents the mean arrival rate of the chain. nodes

The service rate μ_i of the queue represents the transmission rate of the node i , as shown in Fig. 3.1.

The queue lengths of the nodes of the network at time instant t is represented as a queue length state vector $\mathbf{q}(t) = q_1(t), \dots, q_i(t), \dots, q_N(t)$, where $q_i(t)$ is the queue length of node i at time instant t . I assume that the nodes are symmetrical, they have the same generation and service rate.

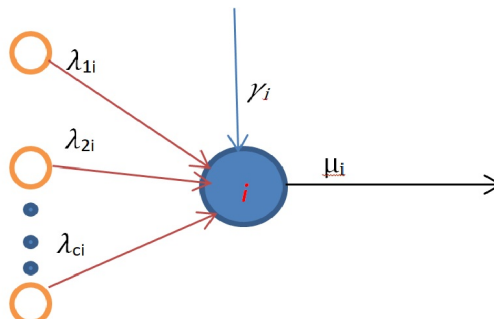


Figure 3.1: The node as a queue

As the case in most of WSNs, I assume that the network is controlled by a wake-up/sleep (ON/OFF) scheme to achieve better energy efficiency. I selected scheduled rendezvous wakeup mechanisms; all the nodes wake up together for the same period, they also return to sleep mode together for the same period too. This mechanism is used in IEEE 802.11 power saving mode (PSM), it is appropriate for a single hop network and when all the nodes are accessible for each other [58].

Theoretically, based on Jackson networks [59, 60], The total mean packet arrival rate to the queue of node i is:

$$\lambda_i = \gamma_i + \sum_{j=1}^c \lambda_j \quad (3.1)$$

$$\lambda = \gamma(I - R)^{-1} \quad (3.2)$$

where I is the identity matrix, R is routing matrix, where R_{ij} is the probability that a packet serviced by node i is sent to node j , and c is the length of the chain. If is average sleep period is β and α is the average wake up period, then the actual mean of arrival and service rates are

$$\lambda_i = \frac{\beta}{\beta + \alpha} \lambda_i \quad (3.3)$$

$$\mu_i = \frac{\beta}{\beta + \alpha} \mu_i \quad (3.4)$$

3.4 Routing with reliability and energy balancing

The back-pressure algorithm was introduced in [53], it deals with both routing and scheduling (forwarding) processes, in routing process the most effective path is defined, but in the scheduling process, the decision to activate the proposed route is taken. At time slot t , back-pressure algorithm calculates the weight of the all possible outgoing links; it defines the link with maximum weight at time slot t as:

$$\omega_{i,j}(t) = \max(\Delta Q_{i,j} - K * \theta_{i,j}) \quad (3.5)$$

where $\Delta Q_{i,j}$ is the differential backlog for both ends of the link, $\theta_{i,j}$ is the cost function and K is a constant used to normalize the cost function, the tie is broken arbitrarily.

The link with maximum weight is activated under schedule $\pi(t)$ based on the following optimization function :

$$\pi(t) = \arg \max_{\pi \in l} (\omega_{i,j}(t) \bar{r}_{i,j}) \quad (3.6)$$

where $\bar{r}_{i,j}$ is the expected link rate and l is the set of all feasible schedules subject to link interference model, and $\omega_{i,j} > 0$ [11, 12]

There are attractive advantages of back-pressure algorithms such as, The optimality of the network throughput (the rate at which the base station receives the packets.), simplicity, adaptive resource allocation, and supporting of stateless and agile routing and scheduling [54].

But also there are some disadvantages such as, complexity (maintaining a large number of queues) and it may cause high E2E delay. A lot of researches have been carried out to improve the performance of the back-pressure algorithm in different network environments [12].

In this chapter, I extend the cost function of the backpressure algorithm with a term depending on how uniform is the distribution of the residual energies of the node, the residual energy is calculated subject to a predefined data loss percentage (ε).

In the previous chapter [1], I showed that optimal path is the one over which a packet is sent to the BS where the minimum residual energy is maximum subject to the constraint that the packet will reach the BS successfully with a given probability, which means that the distribution of the residual energy falls close to uniform.

Thus, the path over which the packet is forwarded to the BS (denoted by \mathfrak{R}) is optimal if:

$$\mathfrak{R} : \max_{\mathfrak{R}} \min_i c_i(t+1) \quad (3.7)$$

subject to the constraint $P_{ij} = \sqrt{1 - \varepsilon}$

I also introduced an entropy-like measure on the energy distribution, by

which we can measure the uniformity of residual energy distribution. At instant $t = 0$, all the nodes have the same level of residual energy, so, the entropy is maximum. We have to keep it as maximum as possible to get closer to uniform residual energy distribution, which means that we need the minimum change in the entropy. The entropy of the current energy distribution is:

$$\mathcal{H}(\bar{\mathbf{c}}) = \sum_{i=1}^N \frac{c_i(t)}{\sum_{j=1}^N c_j(t)} \log \frac{\sum_{j=1}^N c_j(k)}{c_i(t)}. \quad (3.8)$$

So, the gradient of the entropy is:

$$\bar{\mathbf{V}} = \nabla H(\bar{\mathbf{c}}) = \frac{\partial H}{\partial c_1}, \frac{\partial H}{\partial c_2}, \dots, \frac{\partial H}{\partial c_N} \quad (3.9)$$

By solving the above equation, we can calculate the gradient of the entropy of the residual energy distribution on the relay nodes and the reset of nodes of the path, so, the changes of the entropy is [1]:

$$\Delta \mathcal{H} = \bar{\mathbf{V}}^T * \Delta \bar{\mathbf{c}}(t). \quad (3.10)$$

where V is the gradient of the entropy, and Δc_i is: $c_i(t+1) - c_i(t)$ and $c_i(t+1)$ is calculated based on Eq.(2.1) and subject to predefined probability of successful packet transmission from node i to node j , $\mathcal{P}_{ij} = \sqrt{1 - \varepsilon}$.

$$c_{i,j}(t+1) = c_{i,j}(t) - \left(-d_{ij} \frac{\theta \sigma_Z^2}{\ln(P_{ij})} \right) \quad (3.11)$$

In my algorithm for WSNs with stochastic packet generation, I propose back-pressure based algorithm, but the weight of the link will depends on the differential backlog for both ends of the link, and the change of the entropy of residual energy of the chain. The cost function $\Theta_{i,j}$ in equation 3.5 will be the change of the entropy $\Delta \mathcal{H}$, the weight of the link increases as the differential backlog increases and the change of the entropy decreases, Equation 3.5 becomes:

$$\omega_{i,j}(t) = \max(\Delta Q_{i,j} - K * \Delta \mathcal{H}_{i,j}) \quad (3.12)$$

The link satisfies Eq.(3.6) is activated in case of asymmetrical link rates, if all the links have the same rate then Eq.(3.12) determines the activated link.

At the end of each ON, the base station broadcast the vector of network energy status and the vector of queue lengths status, each node uses this information to decide its target. BPEEBP allows multiple instantaneous transmissions. Greedy LQF (longest queue's length first) algorithm is used to schedule transmission links. Since each node has just one transceiver, receivers of higher priority transmitters (higher LQ) will be allocated as interfered and removed from available links. All the links have the same link rate.

3.5 Simulation and performance evaluation

I evaluated the performance of the proposed algorithm by comparing with traditional back-pressure. I used Matlab and Simulink (simevents) to model and simulate the proposed system; each simulation lasts for 1000 units of time. The simulated network consists of 100 symmetrical nodes which are deployed randomly onto a grid of 100X100 according to the 2D normal distribution; the base station is selected randomly, a set of short paths are formed by the Dijkstra shortest path algorithm.

The transmission energy will be calculated according to the Rayleigh fading model Eq.(2.1) under the following assumptions:

- Energy needed by the electronics will be neglecting conditioning.
- All the nodes have $500J$ initial energy.
- $\theta = 10^{-6}$.
- $\sigma_Z^2 = 0.1$.
- $\alpha = 2$.
- P_{ij} is calculated based on reliability $(1 - \varepsilon)$.
- lifespan is the number of packets transmitted over the chain until the first node goes flat.
- The the base station is accessible directly by all the node.
- The the base station appears as a zero-length queue.
- nodes generate packet stochastically with the same average rates γ follow a Poisson distribution with a mean of γ .
- All the nodes have the same service rate and the same link capacity.

I use the variance of consumed energy to express the uniformity of the distribution of consumed energy. Figure 3.2 shows the variance of consumed energy in different chain lengths, 5, 6, 7, 8 and 9. it shows that BPEEBP has lower variance than traditional back-pressure for all lengths of the chains (5,6,7,8,9). Figure 3.3 shows the total consumed energy by the nodes during the simulation period The one can note that all the nodes consumed less energy when using BPEEBP in about 9%

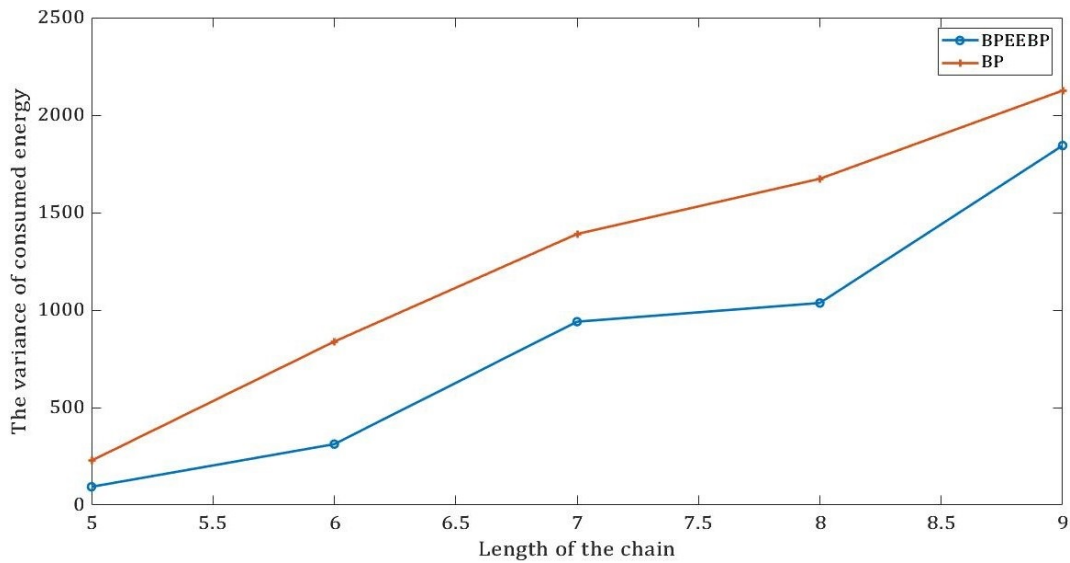


Figure 3.2: The variance of consumed energy VS length of the chain

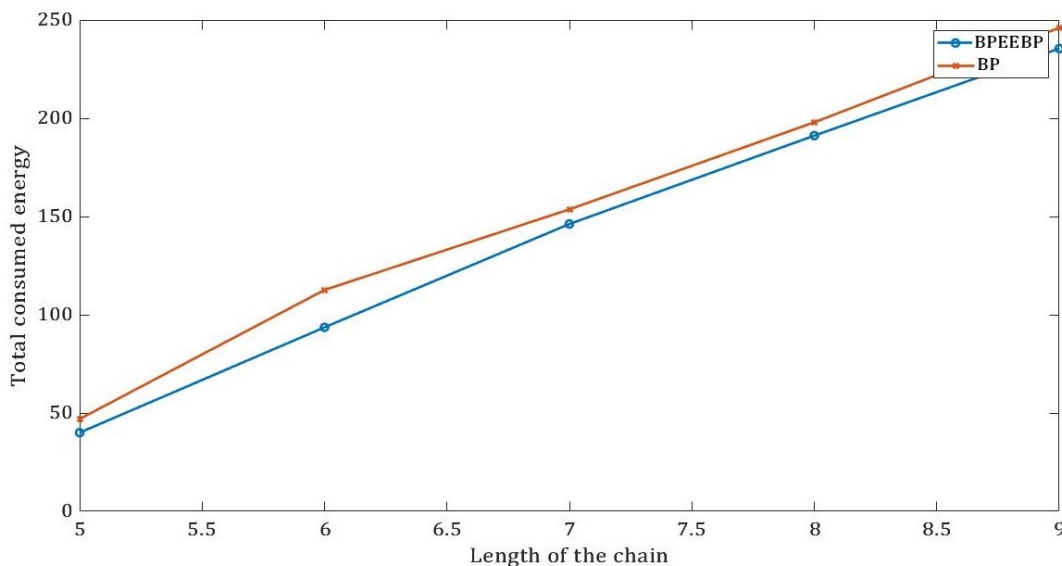


Figure 3.3: Total consumed energy vs length of the chain

In the second experiment, I studied the effect of the utilization of nodes ($\rho_i = \gamma_i/\mu_i$), Figure 3.4 shows the relationship between the variance and the utilization, Figure 3.5 shows the relation between the total consumed energy by the nodes of the chain and different the utilization values 0 – 0.9, both figures show better performance for BPEEBP over the traditional back-pressure .

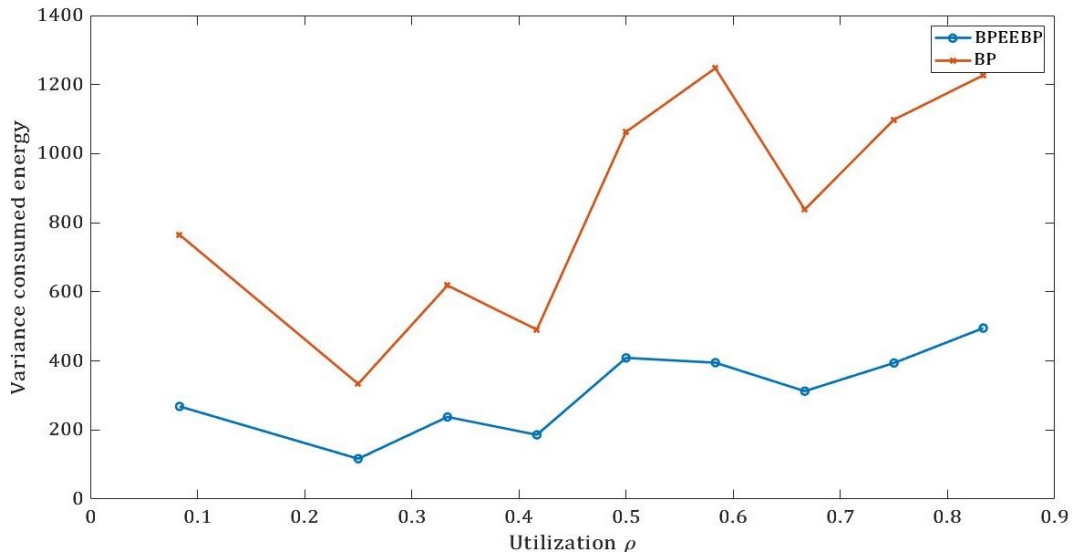


Figure 3.4: Variance of consumed energy vs Utilization ρ

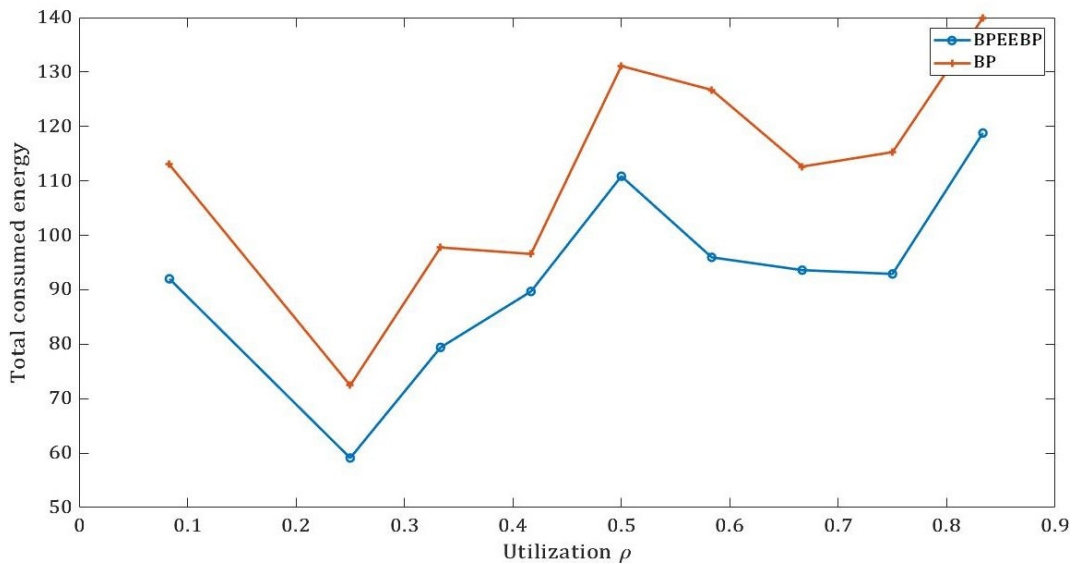


Figure 3.5: Total consumed energy vs Utilization ρ

Figure 3.6 shows that we still have a high packet delivery rate (throughput), there is about 16% enhancement in comparison with traditional pack-pressure by using BPEEBP. The figure shows the relation between the packet arrival rate at the base station and packet generation rate at the nodes.

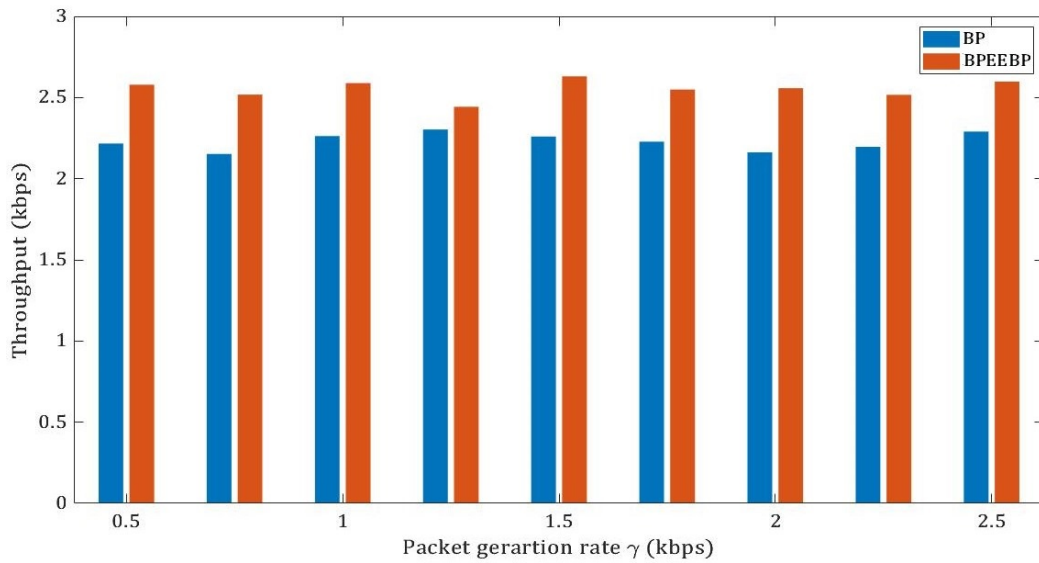


Figure 3.6: Throughput vs packet generation rate γ

Figure 3.7 shows that in the case of BPEEBP, better uniformity of distribution of consumed energy and better load balance among the nodes yields longer lifespan. I studied the lifespan in case of the traditional back-pressure algorithm and BPEEBP regarding different utilization values ρ (0.1-0.9); the figure shows that we have a significantly longer lifespan for all tested cases in about 50% in average.

The average of E2E delay has been enhanced in an average of 15% by using BPEEBP as shown in Figure 3.8.

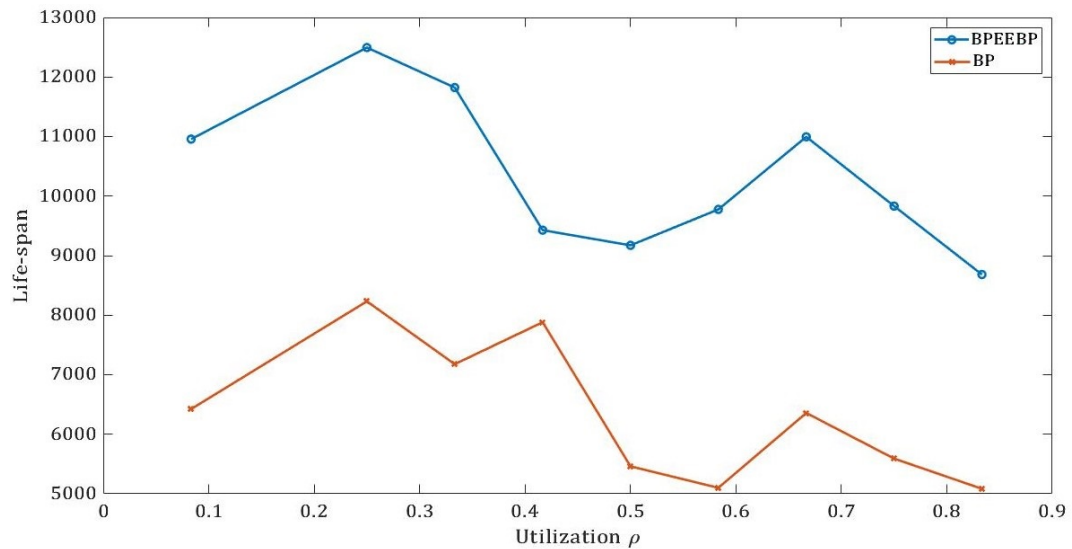


Figure 3.7: Lifespan vs Utilization ρ

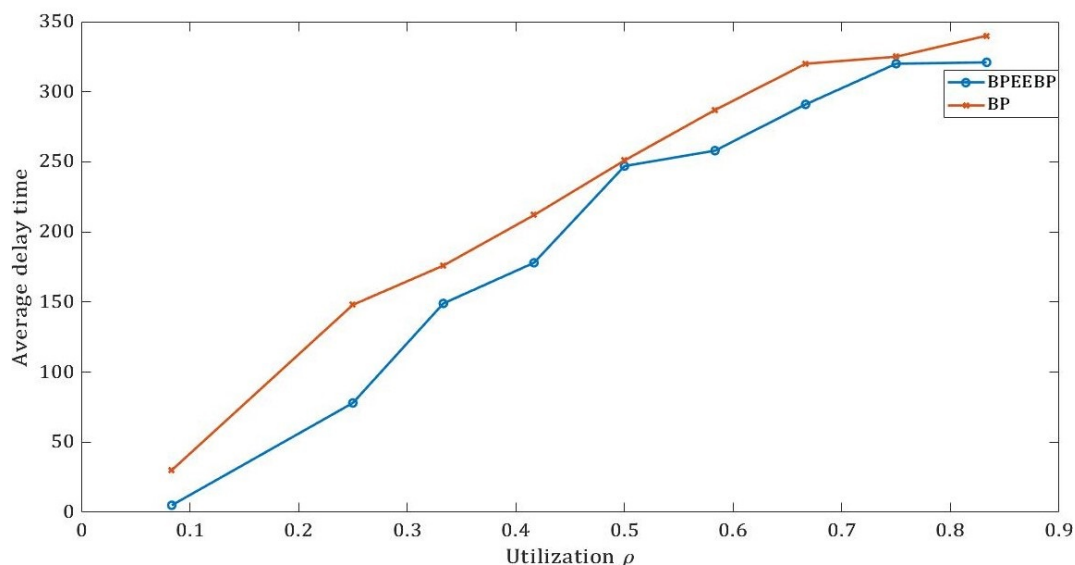


Figure 3.8: Average delay time vs Utilization ρ

The above results ensure that using a cost function based on the entropy of residual energy guarantees the uniform distribution of residual energy among the nodes and guarantees uniform distribution of the traffic among the nodes. Because using the node i as a relay does not depend on its location or energy, it depends on the energy of the network as a whole.

3.6 Conclusions

In this chapter, I proposed a back-pressure and energy entropy based protocol (BPEEBP). Proposed protocol maximizes the energy and traffic balance by minimizing the change of the entropy of the residual energy of the WSNs. It controls the distribution of the traffic and the residual energy subject to a predefined reliability constraint. I compared the performance of BPEEBP to the performance of traditional back-pressure algorithm. BPEEBP shows better performance regarding energy efficiency, E2E delay, and throughput. This work assumes symmetrical nodes regarding packet generation and service rates.

Chapter 4

Reliable and energy-efficient algorithm based on data reduction

In this chapter [3], I developed a data reduction technique to improve the energy efficiency and reliability of WSNs. Compressive sensing (CS) is a data reduction technique used to recover extensive data from fewer samples in case of sparse representation of sensor readings. Unfortunately, energy efficiency and accuracy are contradictory measures as increased accuracy requires a large number of measurements and data transmissions. Therefore, in this chapter, a CS-based algorithm is proposed for efficient data transfer through WSNs, which uses Multiple Objectives Genetic Algorithms (MOGA) to optimize the number of measurements, transmission range, and the sensing matrix. The algorithm aims at striking a good balance between energy efficiency and accuracy. It identifies packet forwarding paths in a multi-hop manner based on the optimized parameters. Numerical simulations and experiments show that the Pareto-front, which is the output of MOGA, helps select the right combination of the number of measurements and the transmission range fitting the application at hand and strike a good balance between energy efficiency and accuracy. The results also demonstrate that the existence of measurement matrices which have lower the mutual coherency improve the accuracy of CS.

4.1 Introduction.

In the traditional Shannon/Nyquist sampling theorem in order to recover the signal correctly, the sampling rate should be higher than twice the highest frequency of the original signal, which generates a vast amount of data. Thus, data must be compressed to reduce the cost of storage and transmission. In 2006; Donoho, Romberg, Candès, and Tao introduced Compressive Sensing

[15, 61, 16, 62], by CS signals can be recovered from fewer samples or measurement under two conditions, the sparsity of the signal (the signal has very few significant coefficients, and the majority are zeros), and the Incoherency among the samples [17]. Recently, a number of researchers have used CS to reduce the amount of data transmitted to the (BS), which improves the energy-efficiency of WSN. CS needs more computational power to reconstruct the signals, but the BSs with their high processing-capabilities and energy can recover sensed signals.

If N is the length of the sparse vector represents the sensed data and M is the number of samples (measurements), then the BS can recover the sparse signal $\mathbf{x} \in R^N$, having only K significant coefficients from the observed vector $\mathbf{y} \in R^M$, $M \ll N$, by solving the linear set of equations $\mathbf{y} = \Phi \mathbf{x}$, where $\Phi \in R^{M \times N}$ is the so-called sensing matrix. Hence, The node performs CS, transmits vector \mathbf{y} and the BS receives it. The energy utilized in the transmission of \mathbf{y} is a function of its length denoted by M . As M increases, more energy is spent; thus we need to reduce the number M to improve the energy efficiency. Unfortunately, the reconstruction error increases as M decreases (for further details, see the next section), which require us striking the right balance between energy efficiency and reconstruction error.

In this chapter, I propose a CS-based chain routing algorithm. The WSN is divided into M paths, the energy efficiency and the reconstruction error are functions of the number of paths, which represent the number of the measurements. Both the energy efficiency and the number of paths depend on the transmission range of the nodes R . The reconstruction error also depends on the quality of the sensing matrix and the number of paths. Finding the optimal number of measurements (M_{opt}), the optimal transmission range of the nodes (R_{opt}), and the optimal sensing matrix (Φ_{opt}) (in terms of mutual coherence) that maximize the energy efficiency and minimizes the reconstruction error is a challenge of optimization. I used multiple objectives genetic algorithms (MOGA) to solve this optimization problem

The problem is treated as follows:

- I proposed a MOGA to strike the right balance between the energy efficiency and reconstruction error of the compressive sensing method.
- I introduced a genetic algorithm to improve the characteristics of the sensing matrix by reducing its mutual coherence.
- I presented a greedy algorithm to split the WSN into multiple paths in a way conserves the balance of the payload, it consists of two sub-algorithms, the first defines the leaf nodes of paths, the second build the paths starting from the leaf nodes

The remainder of the chapter is organized as follows:

- In Section 4.2, I provide an overview of related work in the literature.

- In Section 4.3, I depict the compressive sensing in WSNs.
- In Section 4.4, I depict the MOGA in WSNs.
- In section 4.5, the proposed CCS-MOGA algorithm is described.
- In Section 4.6, I give the numerical results of a detailed performance of the algorithm under different scenarios.
- In section 4.7, I state some conclusions.

4.2 Related works.

Various data reduction schemes are used to enhance the performance metrics of WSNs. Network Coding (NC) is an example of a network-based data reduction technique; research such as [46, 18, 63, 20, 64] use network coding to maximize the life and the reliability of WSNs. They use combined packets as redundant data to improve the E2E error rate.

Compressive sensing is an example of compression-based data reduction techniques. Recently, several algorithms were developed to host CS into WSNs, some of them aim to reduce the payload and the energy dissipated in data collection by shrinking the amount of transmitted data. Some researchers proposed algorithms to improve the reconstructions error; others aim to reduce the complexity of the algorithms or to improve the security of WSNs [21].

In the literature, there are two schemes of CS, plain-CS (all the nodes perform CS) and hybrid-CS (specific nodes perform CS). The authors of [65] suppose hybrid-CS for chain-based WSNs, some nodes forward native packets (Forwarders), others perform CS (Aggregators). Nodes that have less than K descendants are forwarders, the nodes that have aggregator descendants, or more than $K - 1$ descendants are considered as an aggregator where K is the length of the compressed vector (number of measurements).

The method discussed in [66] is also a hybrid-CS algorithm, which is based on mobile agents. There are M agents, where M is the number of measurements, in this paper the authors developed a greedy algorithm to determine the path of the mobile agent. The mobile agent collects the reading of interest nodes; they are the nodes corresponding to the non-zero elements of the row of the measurement matrix agreeing with the ID of the mobile agents. The measurement matrix is a sparse random binary matrix. When all the agents return to the BS, it uses compressive sensing reconstruction algorithms to get the readings of nodes. Unfortunately, the sensing matrix is random, and it may happen that some nodes remain unvisited.

Random walk (RW) routing is used in [67], M random walks are initiated, where M is the number of measurements; in each round, the readings of k random nodes are collected. The authors use the mixing time of the graph to

determine k , the number of nodes to be visited by RW, the k^{th} node collects the sparse measurement of round M_i , then it applies Φ_i of the sensing matrix to produce $\mathbf{y}_i, i = 1, 2, \dots, M$. The measurement vector \mathbf{y} is sent directly to the BS, which collects the M measurement vectors needed for the recovery process.

The algorithm proposed in [68] is a plain-CS for chain-based WSNs; all the nodes are connected like a chain. Each node creates its sensing vector based on a global seed broadcasted by the sink. The nodes forward the summed compressed vector from one to another until the head of the chain, which sends the received vector directly to the sink.

Energy-Efficient Compressive Sensing-Based Clustering Routing Protocol (EECSR) is a plain-CS algorithm [69]; it is a cluster-based scheme. The authors suppose a sector-shaped network, divided into k layers; all of them have the same width. They calculate the optimal number of the clusters for each layer, the optimal size of the clusters, and the optimal location of the cluster heads (CHs) in terms of energy efficiency. All the calculations depend on the geometrical specification of the network and the number of measurements. Members of the clusters transmit compressed to the CHs. CHs collect the compressed data and forward it to BS.

The above mentioned algorithms are based on random measurement matrices; nevertheless, the authors of [22] propose Cluster Size Load Balancing for CS algorithm (CSLB-CS); it is a cluster-based algorithm. Besides using a cluster load balancing technique to reduce the total number of transmissions, they use Chicken Swarm Optimization Algorithm to improve the accuracy and robustness of the reconstruction process.

Some works combine more than one data reduction technique; for example, [70] proposes a compression scheme that combines CS, NC, and spatial-temporal compression. It supposes a clustered topology; the authors formulate an optimization model to minimize the reconstruction error; they presume that the reconstruction error is a function of the total flow rate and link transmission rate. The overall flow rate is measured as the number of received measurements; a predefined reliability probability is considered as an optimization parameter.

Some researches concentrate on improving the energy efficiency [18, 19, 20, 21], others concentrate on improving reconstruction error [22, 70, 23]. In this work, I balanced among payload, energy-efficiency, and accuracy of the reconstruction process by minimizing reconstruction error.

4.3 Compressive sensing in WSNs.

The fundamental idea of CS is that certain signals can be recovered correctly from fewer samples or measurements than classical methods; the possibility of this idea depends on two principles:

1. Sparsity: the signal has only very few significant coefficients, and the

majority of the coefficients are zeros.

2. Incoherency: The coherency between two signals decreases as the linearity between them decreases

If $\boldsymbol{\theta}$ is a dense vector represents the original signal, then it can be expanded into an orthonormal basis $\boldsymbol{\Psi}$ (such as wavelet basis, Fourier basis, DCT, etc.) to acquire a sparse vector \boldsymbol{x} as follows:

$$\boldsymbol{x} = \boldsymbol{\Psi}\boldsymbol{\theta}, \boldsymbol{\Psi} \in R^{N \times N}, \boldsymbol{\theta} \in R^N. \quad (4.1)$$

\boldsymbol{x} as a sparse vector with length N and with K significant element is sampled into vector \boldsymbol{y} , which has the sampled values of \boldsymbol{x} as follows:

$$\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{x}, \boldsymbol{y} \in R^M, \boldsymbol{\Phi} \in R^{M \times N} \quad (4.2)$$

where $\boldsymbol{\Phi}$ is the sensing matrix (Measurement matrix) and M is the number of measurements (samples) and $N \gg M$.

In compressive sensing to reconstruct \boldsymbol{x} correctly from \boldsymbol{y} , $\boldsymbol{\Phi}$ should obey these two conditions:

- The coherence (μ) between $\boldsymbol{\Psi}$ and $\boldsymbol{\Phi}$ should be as minimum as possible where:

$$\mu(\boldsymbol{\Phi}, \boldsymbol{\Psi}) = \sqrt{N} \max_{\substack{1 \leq j \leq M \\ 1 \leq i \leq N}} | \langle \boldsymbol{\varphi}_j, \boldsymbol{\psi}_i \rangle | \quad (4.3)$$

The upper bound of μ is one, where the lower bound is a function of M and N : $\sqrt{\frac{N-M}{M(N-1)}}$

- It satisfies the restricted isometry property (RIP) of order K , which is achieved if the restricted isometry constant (RIC) δ_k is not close to one, where:

$$(1 - \delta_k) \|x\|_{l_2}^2 \leq \|\boldsymbol{\Phi}\|_{l_2}^2 \leq (1 + \delta_k) \|x\|_{l_2}^2 \quad (4.4)$$

Both properties are related to each other

$$\delta_k = (k - 1)\mu \quad (4.5)$$

As μ decreases, δ_k decreases, small δ_k means a higher probability that the sensing matrix satisfies the RIP. The probability of successful data construction depends on the number of measurements M . M is determined based on δ_k and μ [71]:

$$M \geq C\mu^2(\boldsymbol{\Phi}, \boldsymbol{\Psi})K \log(N/\delta) \quad (4.6)$$

This equation indicates that:

- As the coherence decreases, fewer measurements are needed
- As the sparsity increases K decreases, fewer measurements are needed
- Smaller δ means a higher number of measurements is needed.

Fortunately, it is an N-complete problem; however, fortunately, any random matrix drawn from Gaussian, +1/-1, Bernoulli distributions, has a high probability of possessing these properties.

Since $N \gg M$, There are infinitely many possible solutions, each solution represents a possible signal reconstruction, if the RIP holds, l_0 -norm and l_1 -norm can be used to reconstruct the signal if $\hat{\mathbf{x}}$ is the reconstructed vector:

$$\hat{\mathbf{x}} = \underset{\mathbf{y}=\Phi\mathbf{x}}{\operatorname{arg\,min}} \|\mathbf{x}\|_{l_0} \quad (4.7)$$

$$\hat{\mathbf{x}} = \underset{\mathbf{y}=\Phi\mathbf{x}}{\operatorname{arg\,min}} \|\mathbf{x}\|_{l_1} \quad (4.8)$$

Reconstruction by l_0 -norm is accurate, but it is slow because it is an NP-complete algorithm, where l_1 -norm is correct, efficient, and linear programming problem [71].

4.4 MOGA in WSNs.

MOGA is nature-inspired meta-heuristic algorithm which is widely used in WSNs to achieve the optimum of multiple contradictory objectives [72, 73]. These objectives are adjusted simultaneously subject to a set of restrictions. In the case of multi-objectives optimization, there is not a specific definition of the optimal solution, so, instead of a single solution yielded by traditional genetic algorithm (GA), MOGA finds a set of multiple non-dominated solutions, all of them are accepted, the best solution is subjective based on the needs of the designer [74].

MOGA uses a vector of fitness functions $\mathbf{F}(z) = [f_1(z), f_2(z), \dots, f_n(z)]^T$ to find a vector of decision variables which satisfy a set of inequality and equality constraints, these constraints define the viable domain that contains all the acceptable solutions.

MOGA is based on Pareto-Optimality (PO); it uses the fitness functions vector to find Pareto-optimal solutions. As shown in Fig.4.1, Pareto-optimal solutions are the vector $\mathbf{Z} = [z_1, z_2, \dots, z_n]^T$ that contains all the feasible solutions that minimize at least one objective without causing a simultaneous increase in any of the other objectives. Pareto-optimal set or Pareto-front $\hat{\mathbf{Z}} = [\hat{z}_1, \hat{z}_2, \dots, \hat{z}_n]^T$ is the vector of solutions that are not dominated by any other solution in solution space, $\hat{\mathbf{Z}}$ is not dominated by \mathbf{Z} if and only if [74]:

$$\forall i \in i = 1, 2, \dots, n, f_i(z) \leq f_i(\hat{z}) \quad (4.9)$$

and

$$\exists i \in \{1, 2, \dots, n\}, f_i(z) < f_i(\hat{z}) \quad (4.10)$$

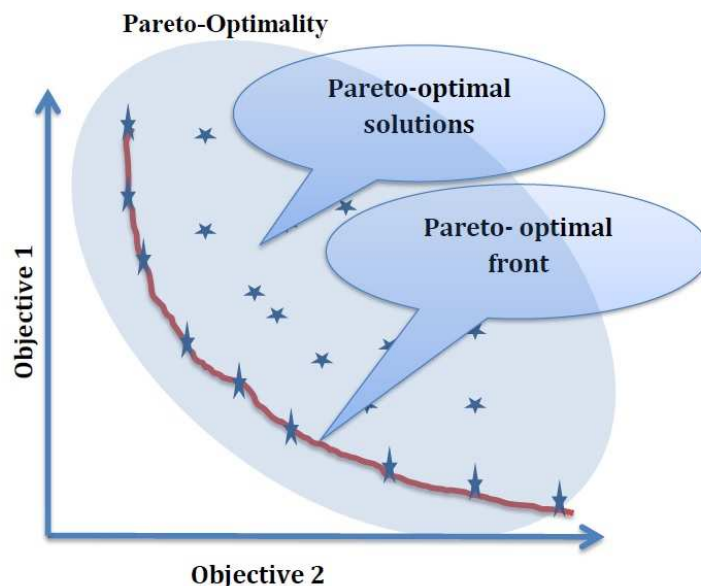


Figure 4.1: Pareto-front.

I developed a multi-objective genetic algorithm that seeks the optimal Number of measurements, the optimal transmission range, and the optimal measurement matrix, based on fitness functions of energy efficiency, reconstruction error, and coherence of sensing matrix. The goal is minimizing the consumed energy as well as the reconstruction error.

In WSN, consumed energy is proportional to d^α where d is the transmission distance, and α is the path-loss exponent. Statistically, if R is the transmission range of the nodes, then the mean square of communication distances $E[r^2] = R^2/2$ [67]. For M paths with n_c is the average path length, the consumed energy of nodes in the path is:

$$E_{ch} = M * n_c * (R^2/2)^\alpha \quad (4.11)$$

If the average distance between the M path leaders and the BS is d_{av} ; then the total energy consumed by the path leaders to transmit a unit of compressed data is:

$$E_{LBS} = \sum_{i=1}^M d_{av}^\alpha \quad (4.12)$$

so, if the sensing field is a square with an edge of length D and the BS is allocated in the centre, the average distance between the chain leaders and the base [67]:

$$d_{av} = \int_0^D \int_0^D \left[\left(x - \frac{D}{2} \right)^2 + \left(y - \frac{D}{2} \right)^2 \right] f(x, y) dx dy \quad (4.13)$$

where $f(x, y)$ is the joint probability function (*pdf*), which is equal to $1/D^2$.

The total energy consumed by the nodes in the path and the path leaders is $E = E_{ch} + E_{LBS}$ from the Eqs.(4.11-4.13), the consumed energy in the network is:

$$E = M \left(n_c \left(\frac{R^2}{2} \right)^{\alpha/2} + \left(\frac{D^2}{6} \right)^{\alpha/2} \right) \quad (4.14)$$

Equation (4.14) shows that as M increases, the consumed energy increases.

CS paradigm has three types of reconstruction errors, original dense data error(e_θ), sparse data error(e_x), and observed data error (e_y). They are defined as follows [23]:

$$e_\theta = \frac{1}{N} \|\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}\|_2^2 \quad (4.15)$$

$$e_x = \frac{1}{N} \|\mathbf{x} - \hat{\mathbf{x}}\|_2^2 \quad (4.16)$$

$$e_y = \frac{1}{M} \|\mathbf{y} - \hat{\mathbf{y}}\|_2^2 \quad (4.17)$$

where $\boldsymbol{\theta}$, \mathbf{x} , and \mathbf{y} are the sensed vectors, and $\hat{\boldsymbol{\theta}}$, $\hat{\mathbf{x}}$, and $\hat{\mathbf{y}}$ are the reconstructed vectors. Authors of [23] prove that the three errors are harmonious, and minimizing one minimizes the others.

In this study, I minimized (e_y), in the same manner as Eq. (4.2), $\hat{\mathbf{y}} = \boldsymbol{\Phi} \hat{\mathbf{x}}$, from Eq. (4.1), $\hat{\mathbf{x}} = \boldsymbol{\Psi} \hat{\boldsymbol{\theta}}$, if $\boldsymbol{\Omega} = \boldsymbol{\Phi} \boldsymbol{\Psi}$, then Eq. (4.17) becomes:

$$e_y = \frac{1}{M} \|\mathbf{y} - \boldsymbol{\Omega} \hat{\mathbf{x}}\|_2^2 \quad (4.18)$$

Eq. (4.18) Shows that as M increase e_y decreases, the fact that contradicts the statement of Eq. (4.14).

To guarantee graph connectivity [75], transmission range R should satisfy $R = \sqrt{A(\log N + \epsilon \log n) / \pi N}$ where A is the area of the sensing field, n is the number of nodes, and ϵ is a constant depends on the structural properties of the graph. The transmission range R and required transmission power P are exponentially proportional [75],

$$P = \frac{R^\alpha (1 + k * \alpha)}{C} \quad (4.19)$$

where R is the transmission range, k size of data unit in the bet, and C is a constant depends on antenna gain, wavelength, and transmission rate and noise level. Using small transmission ranges improves energy efficiency, but more paths are needed to guarantee connectivity, which means a larger M .

Each sensor node standard supports several levels of transmission levels.

For example, MICA2 has 26 transmission power levels from 0.01mW (corresponds to 28m transmission range) to 3.1623mW (corresponds to 118.1.3m transmission range) [76].

Each node uses two transmission ranges:

- R_c : it is a narrow range; the nodes use it to communicate with their neighbors of the same path; it is small because it covers just the average distance between two nodes. We have to ensure that the degree of each node ≥ 2 , one for receiving and the other for transmission.
- R_{BS} : it is broader than R_c ; the path leader nodes use this range to send the compressed data to the BS directly.

4.5 CCS-MOGA

I supposed two types of rounds (as they are introduced in [77]): transmission rounds and construction rounds, which is used in dynamic topology WSN, reconstruction round appears in the case of the mobile sink, mobile nodes, or in case of energy holes problem.

In my proposed model, transmission rounds appear deterministically when nodes have to collect data. Construction rounds appear in the initialization stage, and when a predefined percentage of the nodes die, transmission rounds appear deterministically when nodes have to collect data. At the beginning of each construction round, the BS uses the multi-objectives genetic algorithm to calculate the optimal number of the measurement M_{opt} ; the optimal transmission range R_{opt} , and the optimal sensing matrix $\Phi_{opt} \in R^{M \times N}$. BS constructs M_{opt} paths using the algorithms in the next section, and broadcasts Φ_{opt} over the network. Each node has a copy of the row number m of $\Phi_{opt}(\phi_m)$.

In this section, I elaborated the path based CS algorithm, optimized by MOGA (CCS-MOGA); it consists of three stages; (i) Seeking M_{opt} , R_{opt} , and Φ_{opt} ; (ii): Construction the network based on optimal values of the MOGA variables; (iii) The accomplishment of CS. The aim is to improve the clashed objectives; energy-efficiency and reconstruction error simultaneously.

The objective functions of MOGA are shown in Eq. (4.14) and Eq. (4.18), subject to the constraints shown in Eq. (4.3), Eq. (4.5), and the specification of MICA2 mentioned in section 4.3. The output of the multiple-objectives genetic algorithm is an optimal measuring matrix with minimum mutual coherency and a Pareto front as shown in the Fig. 4.2. The figure shows that as the energy increases the reconstruction error decreases and vice versa. Table (4.1) shows these optimal values of the energy and the reconstruction error, it also shows the corresponding values of M and R .

The optimal point of Pareto front is the knee point of the curve in Fig. 4.2, but the user can use another point up to his/her concerns, in the direction of energy efficiency or accuracy.

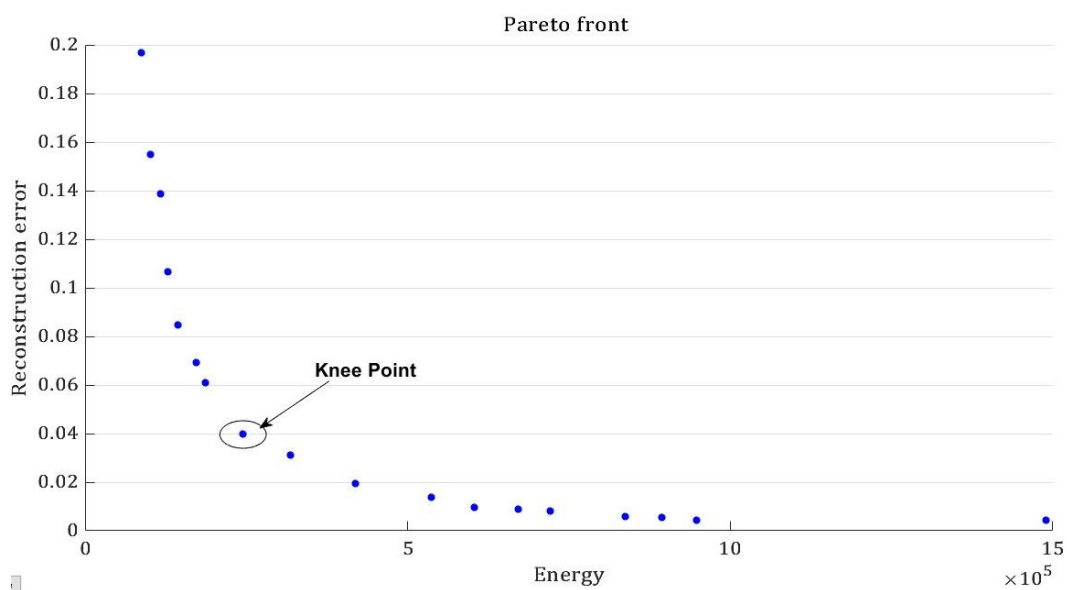


Figure 4.2: Pareto-optimal front.

Table 4.1: Pareto-optimal objectives and variables.

Energy	Error	M	R
947916.18	0.00	117.00	36.65
1489596.81	0.00	124.00	72.82
87843.64	0.20	5.00	20.88
836503.57	0.01	108.00	30.53
116910.65	0.14	9.00	21.34
243995.40	0.04	27.00	22.63
670631.02	0.01	90.00	23.77
893440.23	0.01	123.00	24.24
602962.65	0.01	82.00	21.22
186960.36	0.06	19.00	21.96
720303.64	0.01	98.00	23.15
127977.69	0.11	11.00	20.91
536020.04	0.01	67.00	26.74
144284.48	0.08	13.00	21.47
418571.96	0.02	52.00	23.98
101220.86	0.15	7.00	20.89
317849.99	0.03	37.00	23.86
172654.17	0.07	17.00	21.78

In the first stage, Besides M_{opt} , R_{opt} , the algorithm optimizes the sensing matrix by minimizing μ ; it starts with a random pattern that follows i.i.d. Gaussian distribution, then it is improved until the value of the fitness function

for the best point in the current population is less than or equal to fitness limit. I defined the fitness limit as the lower boundary of the μ .

Based on calculated M_{opt} and R_{opt} the WSN is divided into M paths, to guarantee maximum coverage and minimum isolated node, I use Algorithm (1) to select M leaf nodes (cl) should satisfy:

$$m \in cl \equiv \max_{\forall m \in N} \{ \min_{\forall i \in cl} dis(m, i) \} \quad (4.20)$$

where $dis(m, i)$ is the distance between leaf node m and node i which means that the leaf nodes are the farthest nodes from the BS and the farthest from each other. Figure 4.3 shows the leaf nodes of WSN consists of 200 nodes deployed in $100 * 100m$ field.

Algorithm 1 Selection of leaf nodes

Ensure: $cl \leftarrow$ vector of path leaves
 $N \leftarrow$ Number of node
 $M \leftarrow$ Number of paths
 $d \leftarrow$ Distances matrix
 $cl(1) \leftarrow BS$
 $cl(2) \leftarrow \max\{d_{BS,:}\}$
for $i = 3$ **to** $M + 1$ **do**
 for $j = 1$ **to** N **do**
 if $i \notin cl$ **then**
 for $k = 1$ **to** $lengthof j$ **do**
 $tds(k) \leftarrow d_{i,cl(k)}$
 end for
 $td(i) \leftarrow \min\{tds\}$
 end if
 end for
 $cl(j) \leftarrow \max(td)$
end for

M paths are constructed; as shown in Algorithm (2), each of them starts with a leaf node as a header, then in turn, each leaf node select the closest node within R_{opt} to be a member of it's path, the new members be the new header, and so on. But the arrangement of the nodes in the path is not optimal, the shortest path is $spath_i = \{cl_i, cl_i + 1, \dots, BS\}$ where:

$$spath_i = \min_{path_i} \sum_{j=1}^{j=n_c-1} dis(j, j+1) \quad (4.21)$$

Algorithm (3) is used to optimize the arrangement of the nodes to achieve the shortest paths as shown in Fig. 4.4.

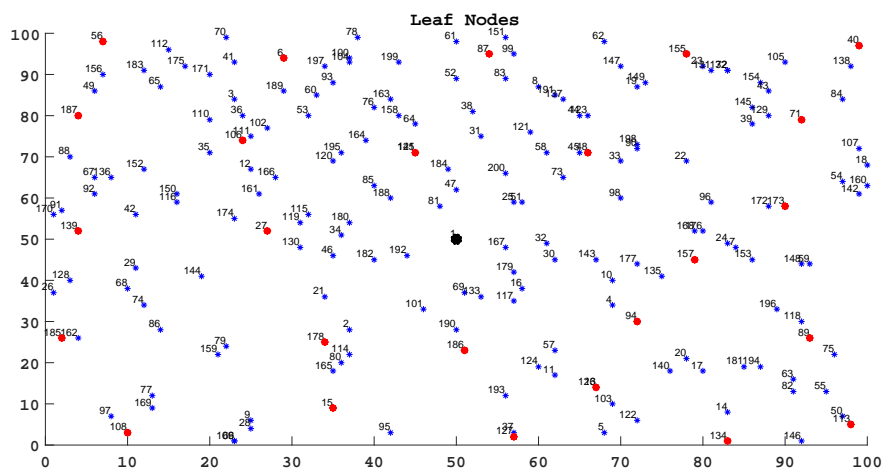


Figure 4.3: Distribution of leaf-nodes.

Algorithm 2 Construction of the paths

Ensure: $paths \leftarrow$ Matrix of the paths.

$cl \leftarrow$ vector of leaf-nodes

$N \leftarrow$ vector of nodes

$M \leftarrow$ Number of paths

$d \leftarrow$ Distances matrix

$cc \leftarrow$ 1 path counter vector

$Headers \leftarrow cl$

$cc = 1$

for $i = 1$ **to** M **do**

$paths(i, cc) = headers(i)$

for $j = 1$ **to** $length(N)$ **do**

$ds(j) \leftarrow d(headers(i), N(j))$

sort ds

$Headers(i + 1) \leftarrow N(\min(ds))$

$N \leftarrow N - (Headers(i))$

end for

end for

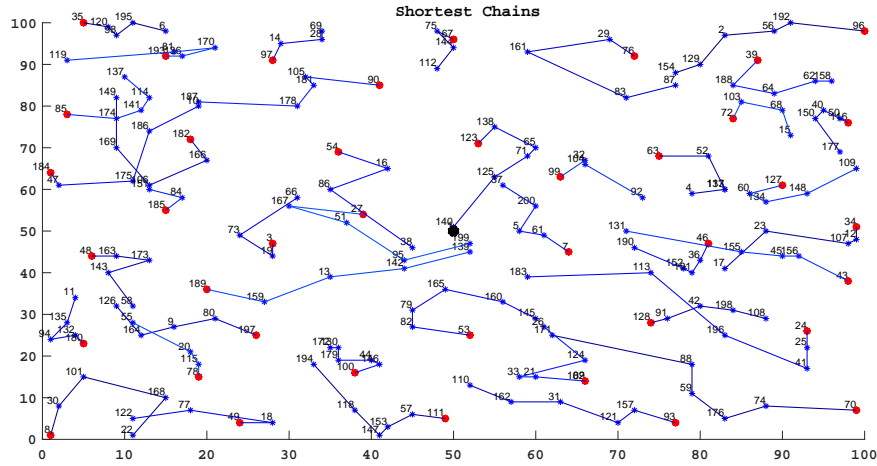


Figure 4.4: Shortest paths.

Algorithm 3 Construction of the shortest paths

Ensure: $spaths \leftarrow$ matrix of the shortest paths.
 $n_c \leftarrow$ path vector
 $s_c \leftarrow$ shortest path vector
 $paths \leftarrow$ array of possible paths
for $i = 1$ **to** $length(paths)$ **do**
 $ds(i) \leftarrow 0$
 for $j = 1$ **to** $length(n_c)$ **do**
 $ds(i) \leftarrow ds(i) + d(paths(i, j), paths(i, j + 1))$
 end for
end for
 $spath = path(\min(ds(i)))$

At the beginning of a transmission round, each path selects a path leader based on the ratio of residual energy and distance from the sink. Each node receives the data from the preceding node and add it to its data, and pass the summation to the next node in the path, $x_i = \sum_{i=1,2,..,n_c} x_i$, $i = \{1, 2, \dots, n_c\}$, and n_c is the length of the path.

CL arranges the received data into $\mathbf{X}_m \in R^{1*N}$; it includes the readings of each node into its corresponding element of \mathbf{X}_m , \mathbf{X}_m is a sparse vector, all its elements are zeros, just the elements corresponding to members of path m , The sparsity level K is determined by the number of measurements, fewer number of measurements mean longer paths and lower level of sparsity. CL calculates $\mathbf{y}_m = \phi_m \mathbf{X}_m$, $m = \{1, 2, \dots, M_{opt}\}$ and transmits it directly to the BS. BS concatenates $y_m, m = \{1, 2, \dots, M_{opt}\}$ to obtain $\mathbf{Y} \in R^{1*M}$ which reconstructed to recover \mathbf{X} , $\mathbf{X} = \cup_{m=1}^M \mathbf{X}_m$. Proposed CS model is demonstrated in Fig. 4.5

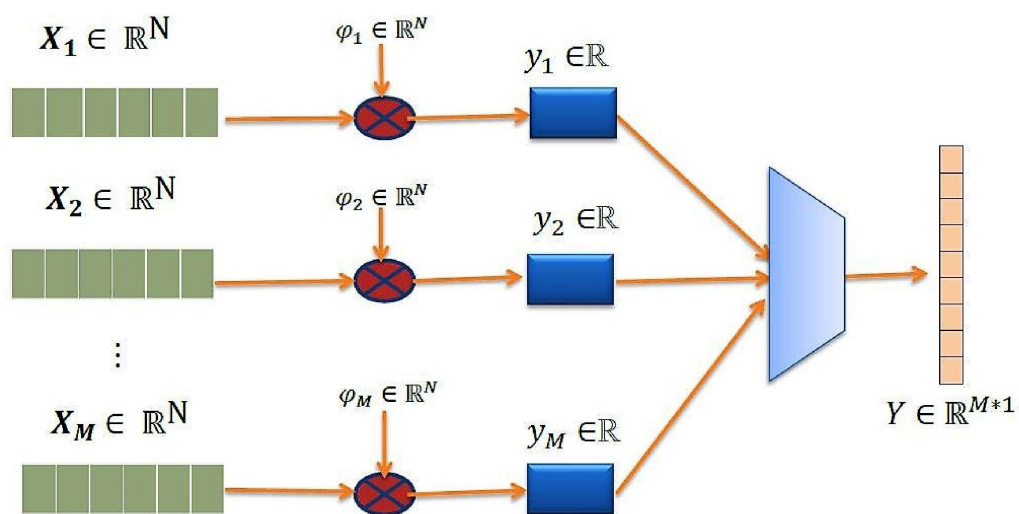


Figure 4.5: Compressive sensing model.

Figure 4.6 shows the flowchart of CCS-MOGA, Figure 4.7 shows the flowchart of CS algorithm in the BS side and Fig. 4.8 show the CS algorithm in the CL side.

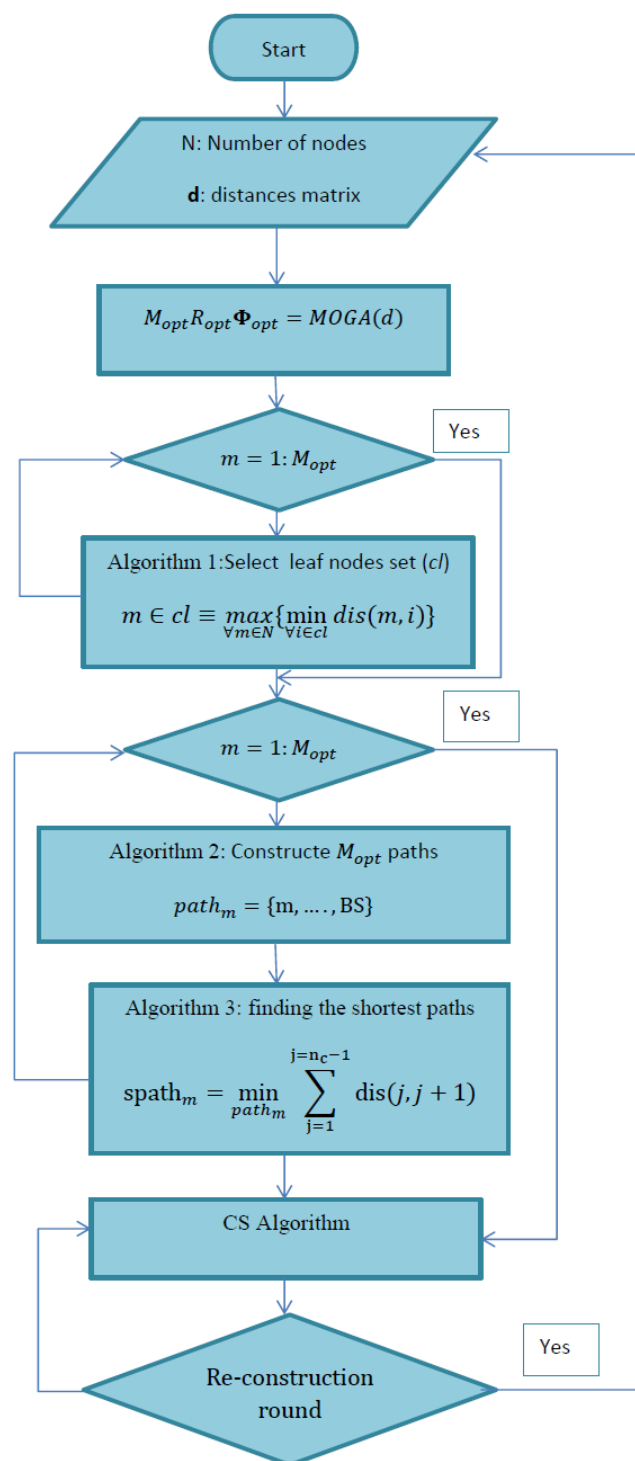


Figure 4.6: The flowchart of CCS-MOGA

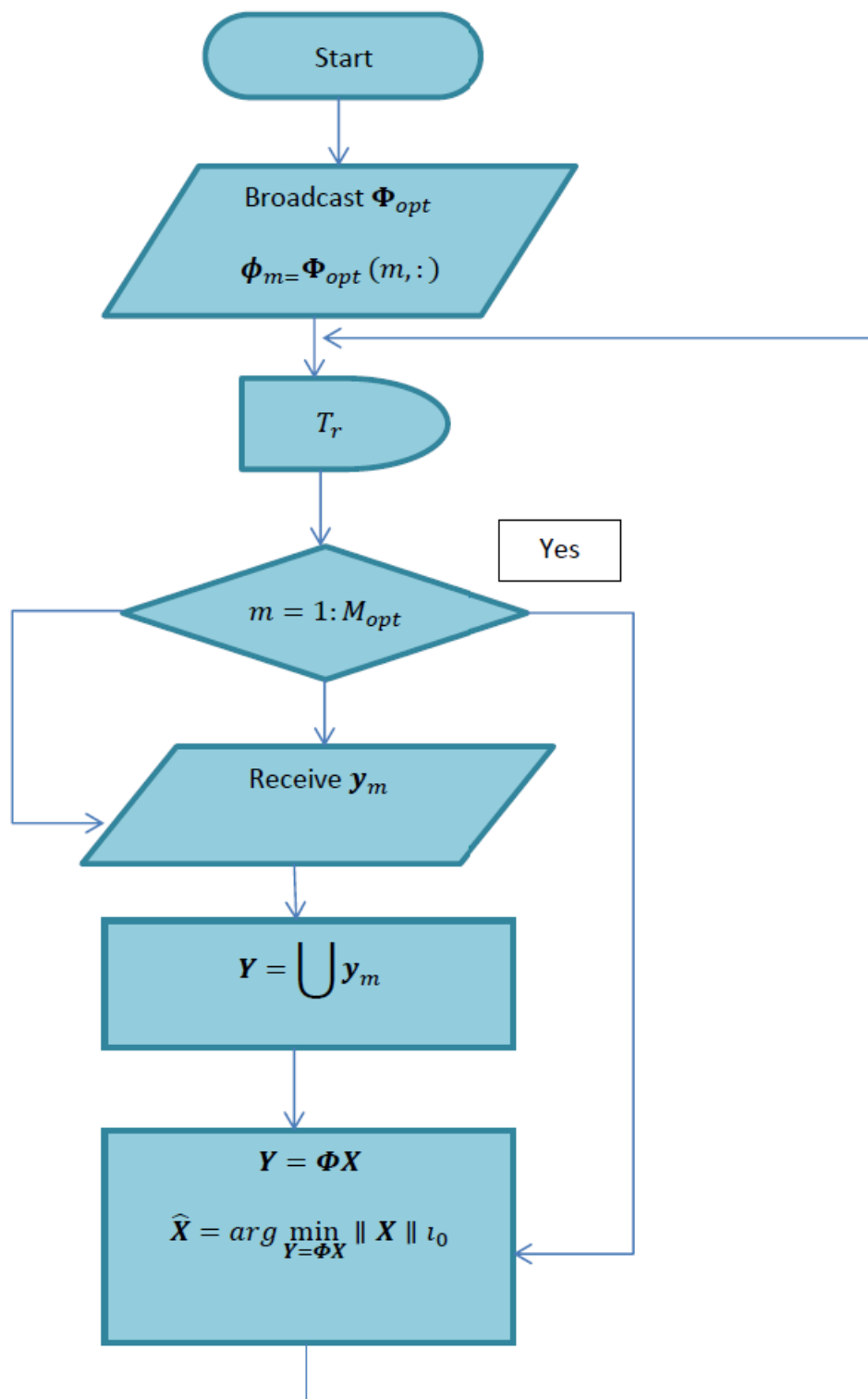


Figure 4.7: The flowchart of CS algorithm in the BS side

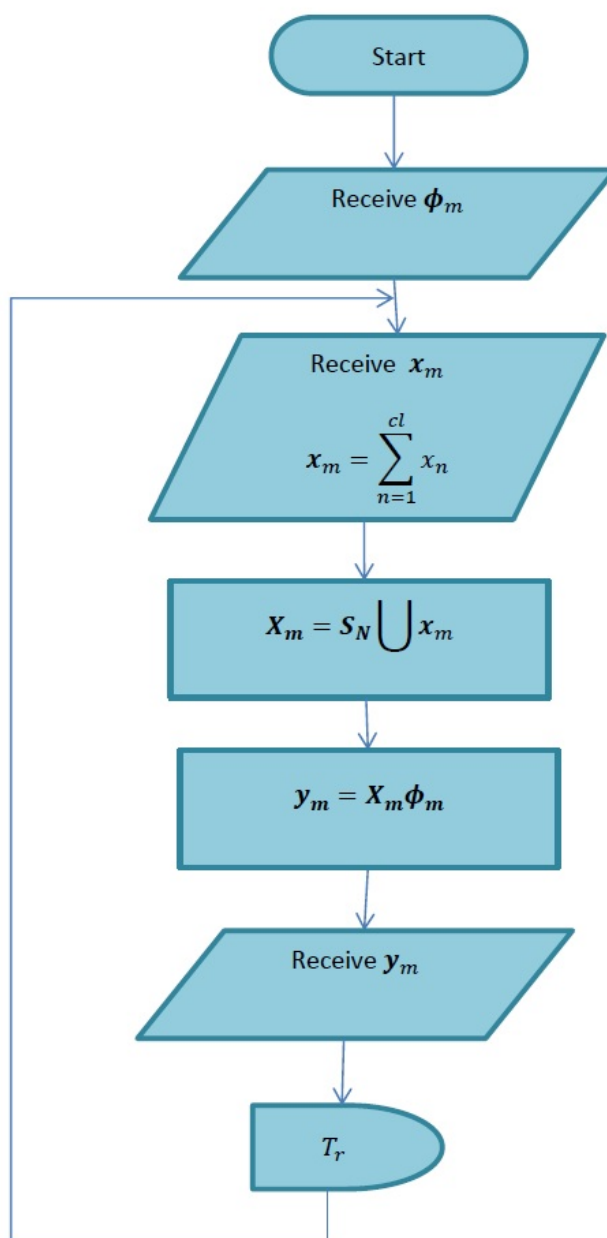


Figure 4.8: The flowchart of CS algorithm in CL side

4.6 Simulation and numerical results

I used MATLAB R2018 to simulate the algorithm. I compared the non-compression, proposed plain compression and some other plain compression algorithms with each other. Also I evaluated the impact of optimization on the system performance with respect to energy efficiency and accuracy of construction.

I assumed different numbers of sensors (25-500 nodes) deployed in a grid

of 100x100m, the nodes are deployed according to a 2D normal distribution. I presumed MICA2, where E_{Tx} is $3.12 \mu\text{J}/\text{bit}$ and E_{Rx} is $2.34 \mu\text{J}/\text{bit}$ [76], the energy spent by the electronics is neglected and α is 2.5 for both short and long transmissions.

In the first experiment, I explored the impact of the optimization of sensing matrix, Figure 4.9 shows the mutual coherency μ for both random and optimized matrices for different sizes of M , it shows that the optimized sensing matrix always has lower μ , which means lower probability of reconstruction error as shown illuminated by the equations (16-18).

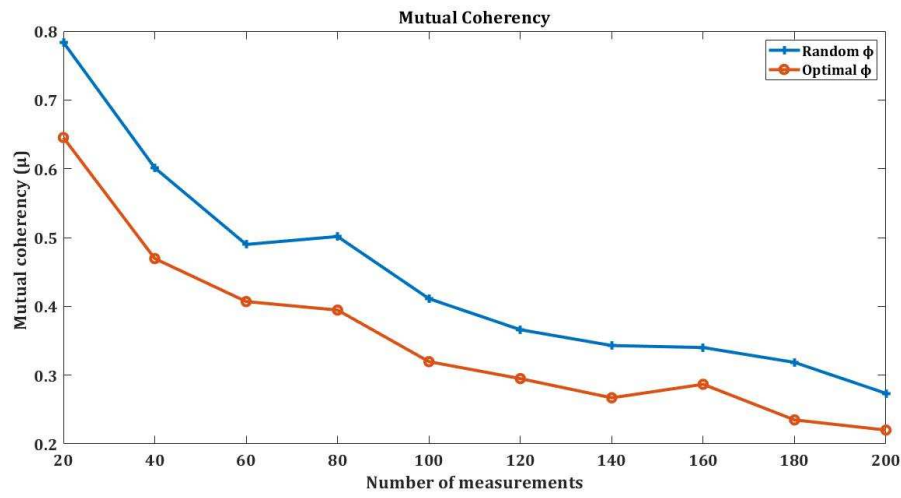


Figure 4.9: Coherency of random and optimal Φ .

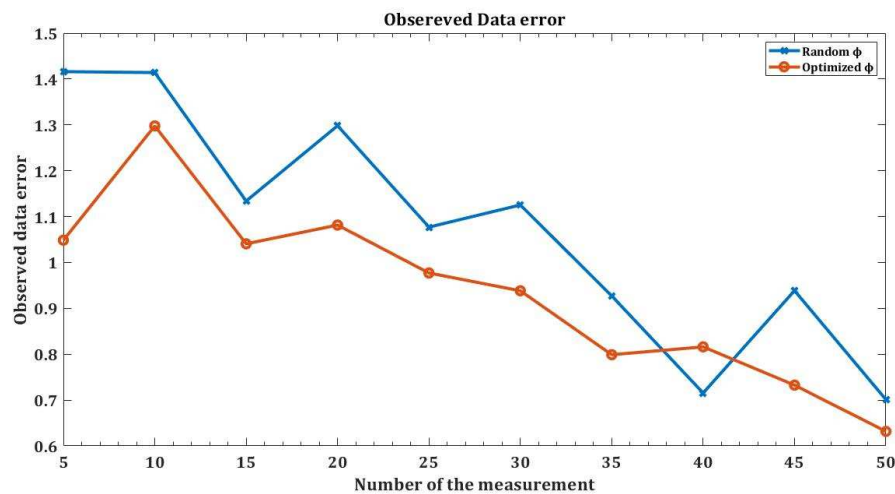


Figure 4.10: Reconstruction error of optimal Φ and random Φ .

Figure 4.10 emphasizes that the optimization of sensing matrix in terms of mutual coherency affords mostly lower reconstruction error regardless of the number of measurements. In some rare case, random matrix may show

low coherence, by the nature of randomness, but we still need a systematic optimization and our systems should not be controlled by coincidences.

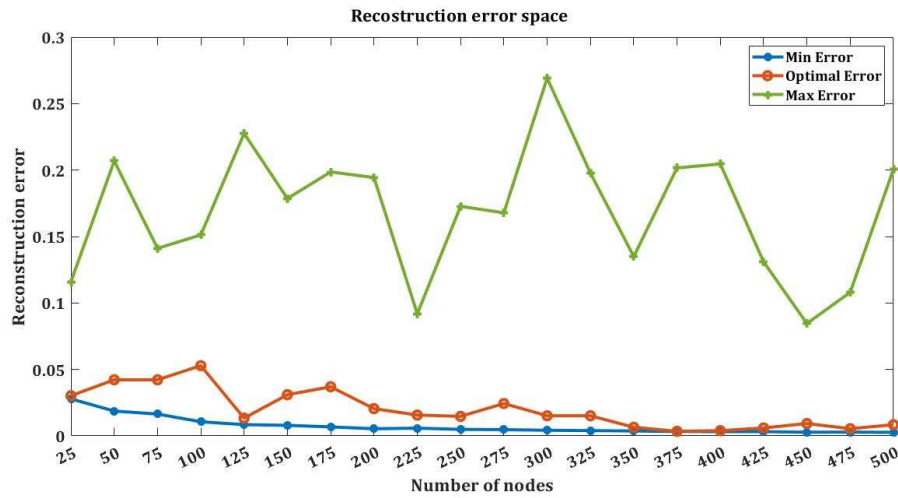


Figure 4.11: The reconstruction error per transmission round.

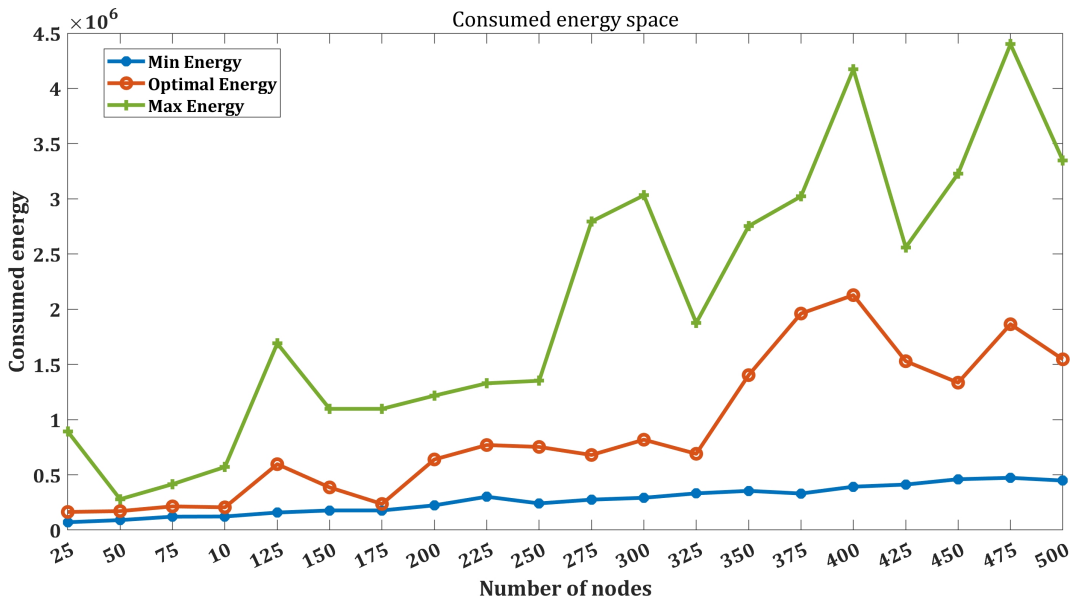


Figure 4.12: The average consumed energy per transmission round.

In the second experiment, I investigated the relations among objectives and variables of optimization system; Figure 4.11 shows the reconstruction error, Figure 4.12 shows the average consumed energy per a round of data collection, I simulated WSNs with different number of sensors.

The figures compare the minimum, the optimal and the maximum values of these two objectives as reported by the Pareto front of the optimization system, the optimal values always found in between minimum and maximum values. One can note an uneven relation between the number of nodes and the

average energy; This occurs because I supposed random distribution of nodes for each case, but in general, there is an increasing tendency of average energy as the number of nodes increases, and the reconstruction error drops as the number of nodes increases.

I assumed a WSN containing 500 nodes for the third experiment. The nodes are deployed in a grid of 100x100m agreeing to a 2D normal distribution. Figure 4.13 shows the relations between the number of measurements as an optimization variable on one side, and the objectives of the optimization from the other side. The left y-axis shows the energy as an objective, where the right y-axis shows the reconstruction error as the second objective.

The result coincides the theoretical depiction mentioned above; as the number of measurements increases more energy is consumed, but less error probability transpires. The crossing point between the two curves matches the knee point of the Pareto front shown in Fig. 4.2.

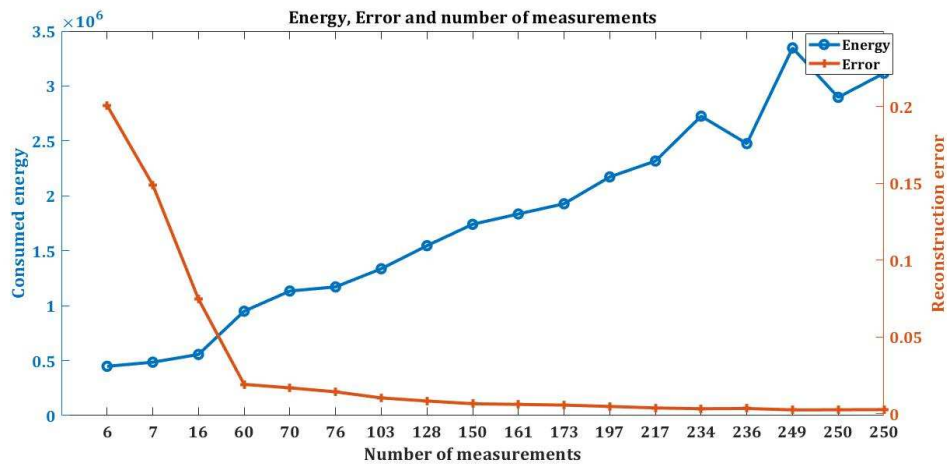


Figure 4.13: The relation among Energy, error and number of measurements of a WSN consists of 500 nodes.

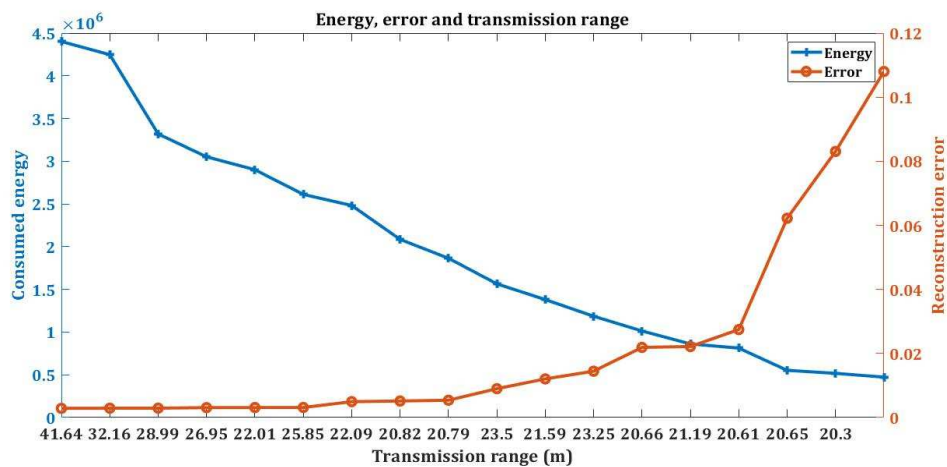


Figure 4.14: The relation among energy, error and transmission rang of WSN consists of 500 nodes.

Figure 4.14 shows the relations among the objectives and the transmission range as a variable. Lower transmission range improves the energy efficiency, exactly as Eq.(4.8) tells, but it increases the reconstruction error because, with low transmission range, less paths are needed to guarantee the connectivity of the WSN, which means fewer number of measurements and longer paths that means lower level of sparsity as explained in section 5.3.

In the last experiments, I explored the energy efficiency of CCS-MOGA as a compressive sensing algorithm. I compared it with two other scenarios:

- the first is the Non-CS algorithm, which represents the traditional multiple path PEGASIS algorithm, where each node fuse the received packet into its packets, then it forwards the fused packets to the closest neighbor, when it arrives at CL, it aggregates the packets together in a predefined packet length (I assumed it 100), then sends them directly to the BS without any type of compression [41].
- the second scenario, I supposed a T-CS algorithm, it is accomplished in many researches such as [67, 68], each CL compresses its data using its own measurements matrix and then sends the samples vector with length of M to the BS.

Figure 4.15 shows the energy efficiency characterized by the average of consumed energy per node after 1000 of transmission rounds. I assumed WSNs with different numbers of nodes (50-500) in the same sensing field of previous experiments. The figure shows much lower energy consumption of CCS-MOGA, it is a reasonable result, because each CL sends just one element of vector \mathbf{y} as shown in 4.5, where in case of non-CS, each CL sends a vector of 100 element. I noted that the average of consumed energy per node in case of T-CS changes linearly with the size of the network, because the length of vector sent by CL depends on M , as M increases the length of the vector increases and the spent energy increases too.

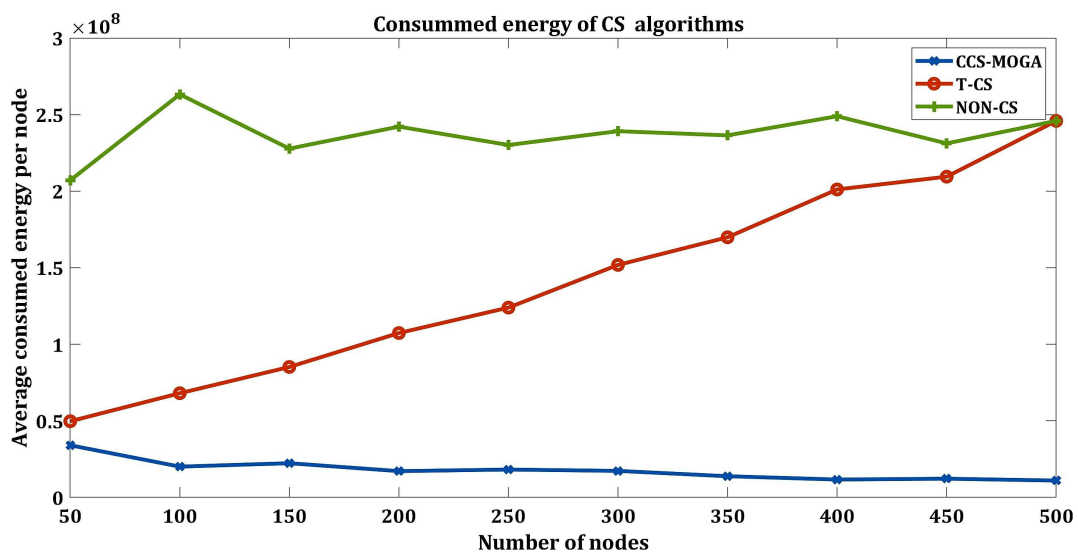


Figure 4.15: The average consumed energy per node.

Energy efficiency is not only measured by reducing the amount of consumed energy, The distribution of the consumed energy is also vital. A uniform distribution of consumed energy among the nodes means the lack of bottleneck nodes and deficiency of overloaded nodes, which may prolong the life span of WSN [1]. Figure 4.16 shows the variance of the consumed energy of the nodes. Low variance means more uniformity of distribution of consumed energy. WSNs use CS-MOGA show a very low variance, so they tend to have longer life span than the other two scenarios.

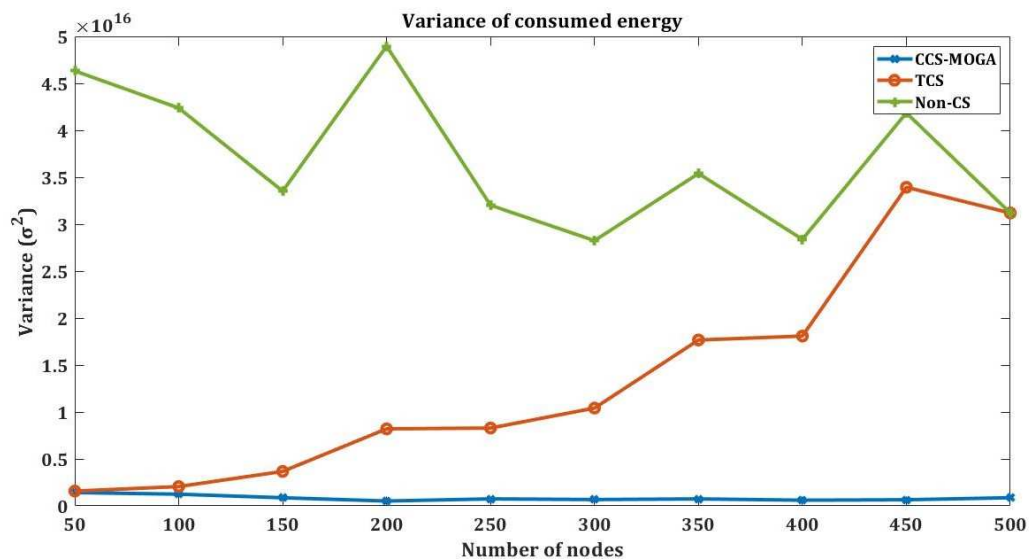


Figure 4.16: The variance of the consumed energy.

In general, CCS-MOGA is a heuristical optimization algorithm, in which the performance of MOGA depends on its parameters (the population size, the maximum number of generation, etc.). To achieve a better performance requires a higher level of parameters, which increases the complexity of the algorithm. Hence, future works are needed to investigate other optimization techniques that improve the performance in terms of optimality and complexity.

4.7 Conclusions

In this chapter, I developed an algorithm based on multiple objectives genetic algorithm to find the optimal values of the variables of compressive sensing paradigm. I optimized the number of measurements, transmission range and the mutual coherence of sensing matrix. I found that the optimizing these variables will indeed maximize the energy efficiency and minimize the probability of reconstruction error. The proposed algorithm provides a good trade-off between these two objectives. I also proposed a compressive sensing procedure that reduces the length of sensing vector to save on energy. The algorithm

provides a dynamic construction of WSN based on the values of optimization variables and objectives . The performance of the proposed algorithms were compared to the performance of multiple paths PEGASIS and traditional CS algorithms and a higher energy efficiency has been achieved.

Chapter 5

Predictive maintenance for WSN

PdM is a new concept of system maintenance that helps system operators evaluate the current status of their systems, and it also assists in predicting the future quality of these systems and scheduling maintenance action. PdMs can significantly increase the lifespan of the system and reduce maintenance costs. In this chapter; I present my PdM model that utilizes machine learning to predict the operational status of the system after M active steps based on K previous observations implemented by a Feedforward Neural Network (FFNN). I used quantization and encoding schemes to reduce the complexity of the system, which enables easy SW implementation. I applied the proposed model to build a PdM system for WSNs, where my concern was to predict the state of the system as far as the quality of data transfer is concerned. The FFNN provides a forward prediction of the operational status of the network based on the previous readings of QoS requirements of WSN (data loss, energy efficiency, throughput, and delay). I also demonstrated the relation between complexity and accuracy (K and M , respectively). I found that larger M leads to longer execution time and larger prediction error, where larger K entails longer execution time and smaller prediction error. I also investigated how quantization and encoding can reduce complexity to implement a real-time PdM system.

5.1 Introduction

Predictive Maintenance (PdM) is concerned with collecting data and estimating the operability of the system under observation. PdM enables the users to evaluate the operating conditions and diagnose anomalies in system behaviour. It may also be used to estimate the time of the next failure and to

approximate the remaining life-time of the system. PdM maximizes the system life cycle and minimizes unplanned downtime, so it also has a significant positive impact on the system's reliability under monitoring and production quality. Furthermore, PdM significantly reduces the cost of maintenance [78].

Wireless Sensors Networks (WSNs) and Internet of things (IoT)[24] technologies are crucial tools used in the development and enhancement of PdM. They enable large-scale data acquisition from sensors distributed on machines, factories, and sites under observation. Effective PdM requires the availability of an active sensing scheme to collect the measurements to describe the working conditions of the maintained systems. The types of sensors and their numbers, distribution, and reliability play a key role in PdM's productivity and quality. The sensing and monitoring process should be continuous, periodic, and remote to guarantee the amount and the accuracy of the data needed for precise prediction and decision [79].

Many researchers and designers use the WSNs and IoT as the backbone data acquisition for their approaches. WSNs provide their solutions with an automatic monitoring system that does not require manual measurements in dangerous and harsh industrial environments. Moreover, wireless communications used with WSNs make it easy to deploy and configure PdM systems. Still, it may suffer from some drawbacks: limited energy resources, security, bandwidth, and limited processing capacity [26].

Besides WSNs and IoT, Machine Learning (ML); and Deep Learning (DL) [80] also are essential tools utilized in the improvement and imperfection of PdM. Neural networks are the foundation of ML/ DL; they accept inputs in a two or one-dimensional form; the output is either a categorical response in the classification model or a continuous response in the case of the regression model. Recently, many ML and DL approaches have emerged, such methods can deal with huge, multi-dimensional, and multi-variate data, and they can realize the relationships within. However, it is essential to use the appropriate approach and develop efficient prediction and classification methods to earn high performance and attain PdM's objectives [81].

In this chapter, I present a new PdM approach consists of a prediction model and ML algorithm. The prediction model estimates the forward probability distribution of the operational status of the monitored system, the information about the monitored system is summarized in a multi-variant time series. The model estimates the probability that the system is still fully operational in the next M steps; it checks that the operability in the next M steps is guaranteed with given reliability determined by predefined parameter ε . The proposed model is implemented by an ML algorithm based on Feedforward Neural Network (FFNN).

This Thesis uses the proposed approach as a PdM for WSNs. The input of PdM is the previous K observations of the QoS parameters; the QoS parameters of the WSN include packet loss (reliability), delay, throughput, and energy consumption; they are represented as a multi-variant times series. The output is a vector that represents the status of WSN after M steps from the

present time instance. We also implement quantization and special encoding schemes to reduce the complexity and memory usage of the model to make it compatible with the limited resources of WSNs.

The remainder of this chapter is organized as follows:

- In Section 5.2, I provide a literature overview of the related work.
- In Section 5.3, I present a formal presentation of the problem and the model.
- In Section 5.4, I customize the model as PdM system for WSNs.
- In section 5.5, I describe the set up of the training data set.
- In section 5.6, I give the numerical results of a detailed performance of the algorithm under different scenarios.
- In section 5.7, state some conclusions.

5.2 Related work

Some researchers credit the invention of PdM to the Rio Grande Railway Company in the '40s of the 20th century [82]. The authors of [78, 83, 84] presented valuable surveys of architectures, approaches, and purposes of PdM systems; they have shown that PdM represents an essential feature of smart manufacturing systems, known as the fourth industrial revolution (industry 4.0). Presently, PdM is a hot research topic in the industry, covering all engineering fields ranging from civil engineering to structural engineering.

The researchers proposed a PdM system in [85] to monitor railway tunnels, wherein [86], the researchers presented a PdM solution for metallic structure against corrosion. Also, in electrical engineering, the author of [87] described how to exploit various technologies to design a PdM system for energy router building equipments; in [88], the researchers used the thermal images and machine learning approach to develop a PdM system for power substation equipments.

There are several techniques of DL /ML utilized in designing PdM systems, most of them implemented by Feed Forwarded Neural Networks (FFNNs). The authors of [89] use FFNN for the prognostics of aircraft gas turbine engines and provide a data-driven model, where the complexity of the model increases with the amount of the collected data.

Each piece of data is related to a different feature of the system under observation, and they reduced the complexity by cutting down on the amount of data by using an appropriate selection of the features and dimension reduction. Data in FFNN models go in one direction, and it has low feature space [80].

The PdM approach proposed in [90] is based on Restricted Boltzmann Machine (RBM) and Support Vector Machine (SVM) algorithms; they used

image-recognition and time series forecasting to classify the collected data as normal or abnormal. It is a fast training model because it consists of just one layer, making it unsuitable for a massive amount of data and noisy environments. Convolutional Neural Network (CNN) model is used in [91]; the authors modified the idea of convolution (used widely in image processing) by adding a Dislocated Time Series (DTS). DTS discovers the relationships among the signals with different intervals in periodic mechanical signals. This technique uses shared weights to make use of neighborhoods, and the output depends on the current observations rather than the previous ones.

The authors of [92] present a survey of research that implemented ML/DL techniques to improve the functionality of WSNs and IoT systems; their central aspect is network intrusion detection. In [93], the researchers used Naïve Bayes (NB) model, FFNN, and Logistic Regression classifier (LR). Their approach predicts the probability of successful reception of the next packet; the inputs of the model are Packet Reception Ratio (PRR), and physical features of previous packets include: Signal to Noise ratio (SNR), Received Signal Strength Indicator (RSSI) and The Link Quality Indicator (LQI).

The authors of [94] proposed an ML model to predict the performance of WSNs in terms of reliability. Their model is based on regression trees, linear regression, and neural networks. The input of the model is a vector of the number of detected nodes (d), Inter-Packet-Interval (IPI), number of received packets (RP), and number of erroneous packets/frames (errP). The output is the estimation of Packet Loss Rate (PLR). In [95], they utilized the Neural Network model (NN) to predict the life-time of sensors based on transmission power level and internode distance. An in-depth learning approach was proposed in [96] to estimate the Energy Consumption (EC) and Packet Delivery Ratio (PDR) depending on ten input features (distance, actual transmissions, queue Size... etc.).

This chapter presents a mathematical analysis of a prediction model for PdMs, and I use it with ML algorithm to build a PDM system for WSNs; most of the studies above use WSNs as the backbone and the key component of PDM [97],[98], to the best of our knowledge, there are very few studies interested in finding PDM for WSN, most of them dominating intrusion detection of IoT systems. In this study, WSN is not only a tool but also the PDM system's subject; the proposed approach takes the QoS and limited resources of WSN into account.

5.3 The system model

This paper proposes a PDM approach consists of :

- prediction model estimates the forward probability distribution of the operational status of the monitored system, the information about the monitored system entered into the model in the form of a multi-variant time series. The model estimates the operational status of the system

during the next M steps; it checks that the operability in the next M steps is guaranteed with given reliability determined by predefined parameter ε .

- ML algorithm to implement the prediction model. The proposed model is implemented by an ML algorithm based on Feedforward Neural Network (FFNN)

5.3.1 Predicting the forward probability distribution by FFNN

Let us assume that the information about the monitored system is summarized in times series $x(k)$, this time series can result from direct measurements or pre-processed data obtained by data fusion. Evaluation on the system state can be summarized as follows:

- If $x(k) > A$ then the system is malfunctioning and urgent maintenance action is required;
- If $x(k) \leq A$ then the system operates normally.

Based on the observations $x(k-1), x(k-2), \dots, x(k-K+1)$ the underlying challenge is to estimate the probability that the system is still fully operational in the next M steps:

$$P(x(k+M) \leq A, x(k+M-1) \leq A, \dots, x(k) \leq A | x(k-1) = i, \dots, x(k-K+1) = j) \quad (5.1)$$

or more precisely, to check whether operability in the next M steps is guaranteed with given reliability determined by parameter ε i.e.

$$M : P(x(k+M) \leq A, x(k+M-1) \leq A, \dots, x(k) \leq A | x(k-1) = i, \dots, x(k-K+1) = j) \geq 1 - \varepsilon \quad (5.2)$$

By introducing the notations:

$$\begin{aligned} \mathbf{x}^+(k) &:= (x(k+M), x(k+M-1), \dots, x(k)) \\ \mathbf{x}^-(k) &:= (x(k-1), \dots, x(k-K+1)) \end{aligned} \quad (5.3)$$

one can write this probability in a more compact form, where set \mathbf{B} is defined as: $\mathbf{B} := \{\mathbf{x} : x_i < A, \dots, x_M < A \forall 1 \leq i \leq M\}$

$$M : P(\mathbf{x}^+(k) \in \mathbf{B} | \mathbf{x}^-(k) = (i, \dots, j)) \geq 1 - \varepsilon \quad (5.4)$$

Introducing the following two vectors

$$\begin{aligned} \mathbf{s}^{(1)} &= (s_1^{(1)}, s_2^{(1)}) = (1, 0) \rightarrow \text{ha } \mathbf{x}^+ \in \mathbf{B} \\ \mathbf{s}^{(2)} &= (s_1^{(2)}, s_2^{(2)}) = (0, 1) \rightarrow \text{ha } \mathbf{x}^+ \notin \mathbf{B} \end{aligned} \quad (5.5)$$

one can form a training set

$$\tau^{(K)} = \{(\mathbf{x}^-(k), s(k)), k = 1, \dots, K\}, s(k) \in \{s^{(1)}, s^{(2)}\} \quad (5.6)$$

I use FFNN to implement the ML algorithm; FFNNs have the most straightforward architecture; they have inputs, outputs, and numbers of hidden layers between them; as the number of hidden layers increases, the data moves in one direction from the input layer to the output layer. This study uses BackPropagation (BP) as a training algorithm; it is one of the most fundamental and common training algorithms. The estimated output is calculated based on the activation function; then, it calculates the estimation error based on the loss function. It goes backward to update the weights based on the gradient of the loss function.

The efficiency of FFNN depends on several factors such as the selection of appropriate activation function, selection of proper training algorithm, the suitable structure of hidden layers, size of the training set, and the accurate description of the problem [99]. Unfortunately, there are no standard rules for selecting, comparing, and testing the solutions; the user's satisfaction in accuracy and complexity is the primary benchmark.

The training set in Eq(6) is used to train the corresponding FFNN, where the input-output mapping of the FFNN is $\mathbf{y} = \text{Net}(\mathbf{x}, \mathbf{w})$, where vector \mathbf{w} refers to the weights of the network subject to learning. The weights can then be optimized by the Back Propagation (BP) algorithms as:

$$\mathbf{w}_{opt} : \min \frac{1}{K} \sum_{k=1}^K \|s(k) - \text{Net}(\mathbf{x}^-(k), \mathbf{w})\|^2 \quad (5.7)$$

Yielding:

$$\frac{1}{K} \sum_{k=1}^K \|s(k) - \text{Net}(\mathbf{x}^-(k), \mathbf{w})\|^2 \rightarrow E \|s - \text{Net}(\mathbf{x}^-, \mathbf{w})\|^2 \quad (5.8)$$

and then:

$$\mathbf{w}_{opt} : \min_w E \|s - \text{Net}(\mathbf{x}^-, \mathbf{w})\|^2 \rightarrow \text{Net}(\mathbf{x}^-, \mathbf{w}) = E(s|\mathbf{x}^-) \quad (5.9)$$

subject to Eq.(5.5)

$$E(s|\mathbf{x}^-) = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} P(x^+ \in B|\mathbf{x}^-) \\ P(x^+ \in B^c|\mathbf{x}^-) \end{pmatrix} \quad (5.10)$$

where:

$$\begin{aligned} E_1(\mathbf{s}|\mathbf{x}^-) &= P(\mathbf{x}^+ \in B|\mathbf{x}^-) \\ E_2(\mathbf{s}|\mathbf{x}^-) &= P(\mathbf{x}^+ \in B^c|\mathbf{x}^-) \end{aligned} \quad (5.11)$$

As a result, after learning, at the output of the FFNN, one can observe the estimated conditioned probabilities once the past observations are given in the input if $P(x^+ \in B | x^-) \geq 1 - \varepsilon$. then there are at least M steps to failure.

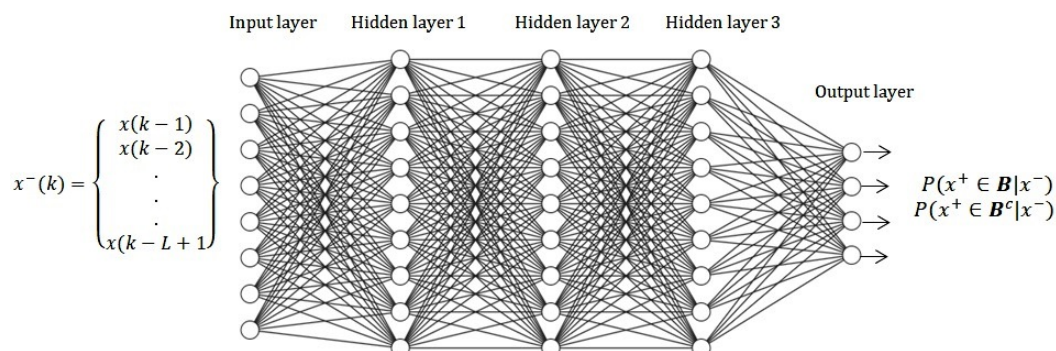


Figure 5.1: FFNN of three hidden layers, and $K=8$ previous observations window.

5.4 "Customized" FFNN for implementing PdM for WSNs – quantization and sparsity

In this thesis, I implemented the PdM approach explained in the previous section to the performance monitoring of WSN. Maintenance procedures of WSNs may include selecting new heads of clusters and leaders of chains, rearrangement of clusters and chains, new sensors deployments, controlling ON/OFF schemes, and many other procedures that enhance the performance of WSNs.

WSNs are limited resource systems in terms of energy, memory, and processing capabilities. The limited resources of WSNs require a low complexity PdM; to reduce the complexity, I use quantization and encoding schemes.

To make our model compatible with such circumstances, I implemented a quantization algorithm to speed up the training process and reduce the complexity of the model. Quantization enhances training speed and complexity, but it weakens the accuracy, so the user has to trade-off complexity with accuracy. Usually, variables and weights are represented as floating-point numbers; the quantization function converts them to integers, fixed-point, or integer numbers; such representations are more efficient regarding memory usage and computation speed [100, 101].

Uniform or deterministic quantization function calculates the quantization

level (q) of the real values r as follows [101]:

$$q(r) = \text{sign}(r) \cdot \Delta \cdot \left\lfloor \frac{|r|}{\Delta} + \frac{1}{2} \right\rfloor \quad (5.12)$$

where Δ is the resolution or the quantization step.

Such functions are known as equidistant quantization. The quantization range is divided equally between quantization levels, so such functions are used in case of uniform distributions of the samples; when the distribution is not uniform, non-equidistant quantization is used; the authors of [97] used Lloyd-Max algorithm to determine the best quantization in such cases, it takes the PDF of samples distribution on account to minimize the mean square quantization error σ . Finding the optimal quantized level q_i of sample r is an iterative process where:

$$q_i(r) = \frac{\int_{c_i}^{c_{i+1}} r f(r) dr}{\int_{c_i}^{c_{i+1}} f(r) dr} \quad (5.13)$$

c_i and c_{i+1} are the regions of the proposed quantization level q_i , and $f(r)$ is the PDF of the samples, the goal is the minimization of (σ), which is :

$$\sigma_q^2 = \sum_{i=1}^Q \int_{c_i}^{c_{i+1}} (r - q_i)^2 f(r) dr \quad (5.14)$$

where Q is the numbers of the quantization levels.

Futhermore, memory is a crucial concern when dealing with FFNN for WSNs; many techniques have been used to improve the memory efficiency of ML/DL algorithms; some of them concern memory requirements of inference, others concern the memory requirements of training. Sparse FFNN is a common and efficient technique used widely to enhance DL/ML algorithms [98].

In sparse FFNN, the input features are represented as a sparse vector; most spare vector elements are zeros, which need fewer computations and less memory space. Besides memory efficiency, sparsity improves the complexity and the computations of the FFNN. Unfortunately, at the same time, it degrades the accuracy of FFNN; the designer has to trade-off between the sparsity level and accuracy[102].

In this study, I used a straightforward encoding scheme used in [97]; it is compatible and complementary with the quantization algorithm, each quantization level is encoded into an orthonormal vector set:

$$\mathbf{q}_l \rightarrow \mathbf{s}\mathbf{q}_l : s_{q_l}(i) = \begin{cases} 1 & \text{if } i = l \\ 0 & \text{otherwise} \end{cases}, i = \{1, 2, \dots, Q\}$$

By the encoding equation (5.3) becomes:

$$\begin{aligned} \mathbf{x}^+(k) &:= (\mathbf{s}\mathbf{q}_{(k+M)}, \mathbf{s}\mathbf{q}_{(k+M-1)}, \dots, \mathbf{s}\mathbf{q}_k) \\ \mathbf{x}^-(k) &:= (\mathbf{s}\mathbf{q}_{(k-1)}, \dots, \mathbf{s}\mathbf{q}_{(k-K+1)}) \end{aligned} \quad (5.15)$$

5.5 Dataset for training

The dataset used for training, validation, and testing is imported from [103]. The researchers collected the data experimentally as described in their paper [104]. They used IEEE 802.15.4 link implemented on TinyOS to connect two TelosB motes; each mote uses a TI CC2420 radio transceiver with 250 kbps. The researchers trace the packet delivery performance under several pre-configured stack parameters; these parameters are related to physical, MAC, and application layers.

Table 5.1: Pre-configured parameters

Parameters	Acronym	Values	Comments
Inter-Arrival Time	IAT (ms)	10, 15, 20, 25, 30, 35, 40, 50	Pre-configured
Packet Payload	PL (bytes)	20, 35, 50, 65, 80, 95, 110	Pre-configured
Maximum Queue Size	QS	1, 30, 60	Pre-configured
Maximum Transmission attempt	NMT	1, 3, 5	Pre-configured
Retry delay	DR	30, 60	Pre-configured
Power of transmission	Ptx	19	Pre-configured
Distance	D	10,20,35	Pre-configured

I generated an observations table consisting of 10000 entries. Each entry summarizes the average measured parameters of 300 packets; I fixed the power transmission level at -19 dBm and change the other pre-configured parameters for the possible combination shown in Table 5.1. Besides the pre-configured parameters, the observations table has several packet delivery performance measured parameters corresponding to each combination of pre-configured parameters, as shown in Table 5.2. A short sample of the observations table is shown in Table 5.3

Table 5.2: Measured parameters

Parameters	Acronym	Values	Comments
Actual Queue Size	AQS	actual values (0–60)	measured
Buffer OverFlow	OF	actual values (0–1)	measured
Actual Transmission attempt	N_A	actual values (0–5)	measured
Actualacknowledged transmission	ACK		measured
Received Signal Strength Indicator	RSSI		measured
Noise Floor	NF		measured
Link Quality Indicator	LQI		measured
Packet arrival time	T_{arr}		measured

Table 5.3: Sample of Observations table

T_{arr}	125304	130758	137716	146187	156155
IAT	10	15	10	15	50
PL	20	35	65	95	110
QS	1	1	30	1	60
NMT	1	1	5	1	5
DR	30	30	30	60	60
P_{tx}	19	19	19	19	19
D	10	10	10	20	35
OF	0	0	0	0	0
Q	0.41	0.23	25.7	0.01	0.08
ACK	0.59	0.77	0.723	0.99	1
N_A	0.593	0.77	0.723	0.993	1.02
$RSSI$	-7.5167	-9.8567	-9.29	-16.31	-22.943
NF	-54.0767	-70.57	-61.0533	-88.9367	-93.71
LQI	63.08	82.3467	77.2833	106.13	106

I used the pre-configured and measured parameters to calculate the QoS requirements of the WSN. Energy efficiency, throughput, delay, and packet loss as in [96, 104].

- *Packet error rate (PER)*: measures the reliability of the system; it depends on the queuing characteristics (Buffering) of the nodes and the quality of the link parameters ($RSSI$, NF , and LQI)

$$PER = \frac{N_A - ACK}{N_A} \quad (5.16)$$

- *Energy efficiency (En)*: determines the energy needed to transmits one beneficial bet; it depends on PER , power transmission level, the payload of the packet, length of the header, and transmission rate:

$$En = \frac{P_{tx} * (PL + PH) * Tt}{PL(1 - PER)} \quad (5.17)$$

PH is the length of the header/trailer, which is (11-31 bytes) in IEEE 802.15.4 [105], Tt is the transmission time which is $0.004ms$ in the case of $250kb/s$.

- *Throughput (Tp)* is the number of beneficial bets received per unit of time; it depends on PL , PER , and transmission service time (Ts).

$$Tp = \frac{PL(1 - PER)}{Ts} \quad (5.18)$$

where :

$$Ts = C + Tt + (N_A * D_R) \quad (5.19)$$

and C is a constant depends on the protocol and the specification of the radio system; it is $\approx 13.5 ms$ in the circumstances of the experiment [104].

- *Delay* is the time elapsed from packet generation to successful packet reception; LQI and queuing characteristics of the nodes are crucial issues when investigating delay. Researchers mostly use queuing system model to state the delay of WSNs; I used system utilization ρ as a metric to quantify the delay, where $\rho = Ts/IAT$ and as $\rho \rightarrow 1$ delay increases.

The four calculated QoS requirements (PER , En , Tp and ρ) are arranged into a $10000 * 4$ input feature table; each entry corresponds to an entry of the observations table. The Packet arrival time (T_{arr}) is reformatted as a time series and added as a fifth column to the input features table.

QoS metrics are contradictory; improving reliability decreases energy efficiency, and improving energy efficiency reduces throughput, and so on; the user should trade-off among these metrics. To define the operational status of the WSN, I defined a range of each metric as follows:

$$\alpha^+ \leq PER < \alpha^-$$

$$\beta^+ \leq En < \beta^-$$

$$\gamma^+ \leq Tp < \gamma^-$$

$$\delta^+ \leq \rho < \delta^-$$

If the four metrics are within the specified range, then the operational state of WSN is “OK” corresponding to $s^{(1)} = (1, 0)$ as defined in Eq. (5.5), which

means that no maintenance is needed; otherwise, the operational status is “NOK” corresponding to $s^{(2)} = (0, 1)$ as defined in Eq. (5.5), which means that maintenance is needed. The operational status for each entry of the input features table represents an entry of the output table of the FFNN, concatenation of the input features table, and the output table forms the dataset of training, testing, and validation of the FFNN. Table 5.4 shows a short sample of the training set.

Table 5.4: Sample of the training dataset

T_{arr}	PER	En	Th	Ru	OP	
					OK	NOK
44488657	0.005618	0.084072	19.98877	3.1304	1	0
44544439	0.003333	0.08388	19.99833	1.0876	1	0
44559087	0	0.0836	20	0.87008	0	1
44597021	0	0.080343	35	1.74016	1	0
44607076	0.006667	0.080882	34.99222	1.450133	1	0

In the next stage, the entries of the training dataset are quantized by the Lloyd-Max algorithm by 8 quantization levels. Each quantized entry is encoded into an 8-bits binary vector, as described in section 4.3. The numerical numbers representing the QoS parameters at instant (t) are converted to a $1*4*8$ sparse vector. Each vector has four 1’s indicate the quantization level of each QoS requirement. Table 5.5 shows a sample of the data set after quantization and encoding.

Table 5.5: Sample of the dataset after quantization and encoding.

T_{arr}	PER	En	Th	Ru	OP
44488657	10000000	00000010	00100000	00010000	10
44544439	01000000	00001000	01000000	00010000	10
44559087	10000000	00100000	00010000	01000000	01
44597021	10000000	00000010	00000100	00000010	01
44607076	00100000	00000010	00000010	00000001	10

5.6 Implementation and simulation results

I implemented the proposed model using the deep learning toolbox of matlab2020b; I used the dataset explained in the previous section.

In the first experiment, I investigated the effect of quantization and encoding on the accuracy and complexity of the PdM system. To get more use of the sparsity of the input vector; the FFNN deals with each binary input vector (as the sample is shown in Table 5.4 as a black and white pattern, where the

ones appear as white points in a black line, figure 5.2 shows a sample of these patterns.

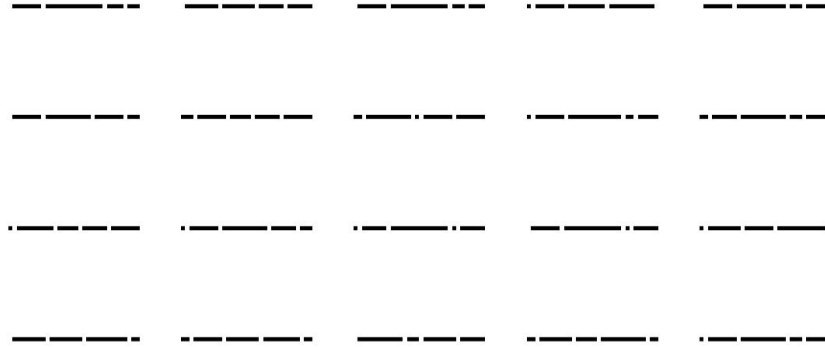


Figure 5.2: Samples of the input vector as black and white patterns.

In this experiment, I used the accuracy as a performance metric, $Acc = R/T$ Where R is the number of correct predictions, and T is the number of the data set. Figure 5.3 shows the complexity of the algorithm under different numbers of hidden layers; I measured the complexity by the execution time of the training process. The figure demonstrates that the algorithm uses quantized and encoded data takes less time than the one raw data, regardless of the number of header layers. The quantized and encoded data ensures better complexity because of the sparsity enlightened in section 4.3. Both algorithms show an ascending tone of training time as the number of hidden increases. The irregularity noticed in both curves is justified by the randomness of initial values of the training process's weight and biases.

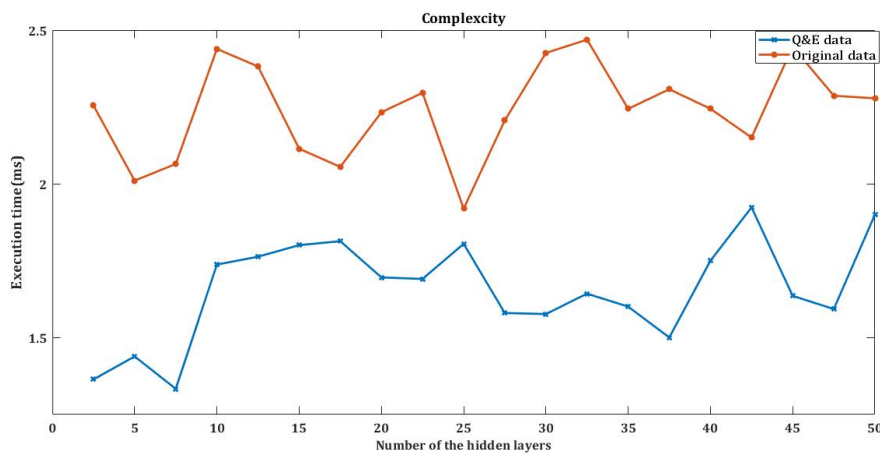


Figure 5.3: Complexity of original data and quantized and encoded data

In Fig. 5.4, one notices that the raw (original) data show better accuracy than the quantized and encoded data; this happens because besides the prediction error, there is also quantization error explained in section 4.2. With quantized and encoded data, the input data appear as a lookup table, so one notices the low variance of accuracy with quantized and encoded data regardless of the number of the hidden layers. The algorithm uses the raw data exhibits better accuracy as the number of hidden layers increases

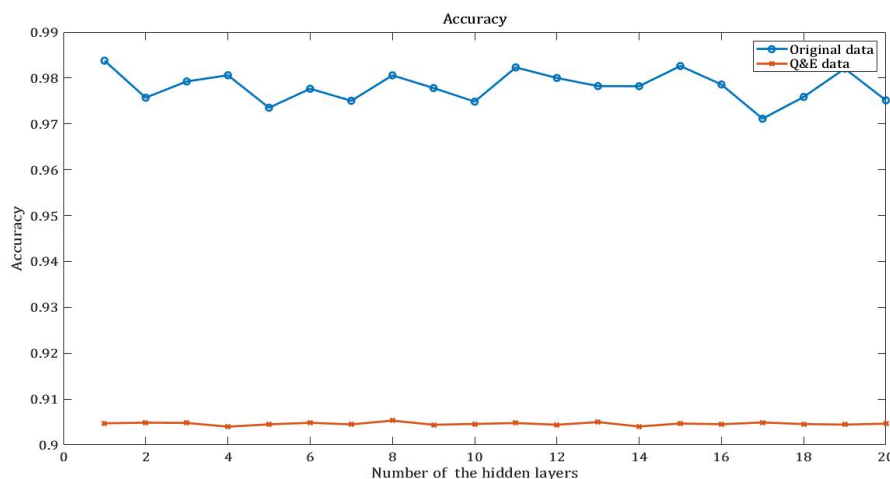


Figure 5.4: Accuracy of original data and quantized and encoded data

In the third experiment, I investigated the relationship between the performance and the number of future time steps M ; two metrics are used to clarify the performance; Mean Square Error MSE and the execution time presenter of the complexity. The output of the FFNN is a binary vector (ops) consists of M elements, the vector (ops) states the operational status of the WSN, $ops(m) = \{m_1, m_2, \dots, m_M\}$,

$$m_i = \begin{cases} 1 & \text{The system will be OK untill step } i. \\ 0 & \text{The system will be fualty after } i \text{ steps.} \end{cases}$$

for example, if $M = 8$, then ops can be $ops = \{1, 1, 1, 1, 1, 0, 0, 0\}$, this means that the system will be faulty after five operational steps, and maintenance should take place.

Figure 5.5 clarifies the performance of the model under different values of $M = (1 - 10)$, where the number of hidden layers is set to ten layers, and the number of previous observations is set to 3. The left y-axis characterizes the MSE , where the right y-axis characterizes the execution time. The figure shows that as M increases, both the execution time and the MSE increase, because longer output vector should be estimated.

Figure 5.6 demonstrates the effect of the number of previous observations k on MSE and execution time. The number of the hidden layer is set to ten, and M is set to 5. The left y-axis represents the MSE , and the right y-axis represents the execution time; a large k means less MSE but a longer

execution time, because longer output vector should be estimated.

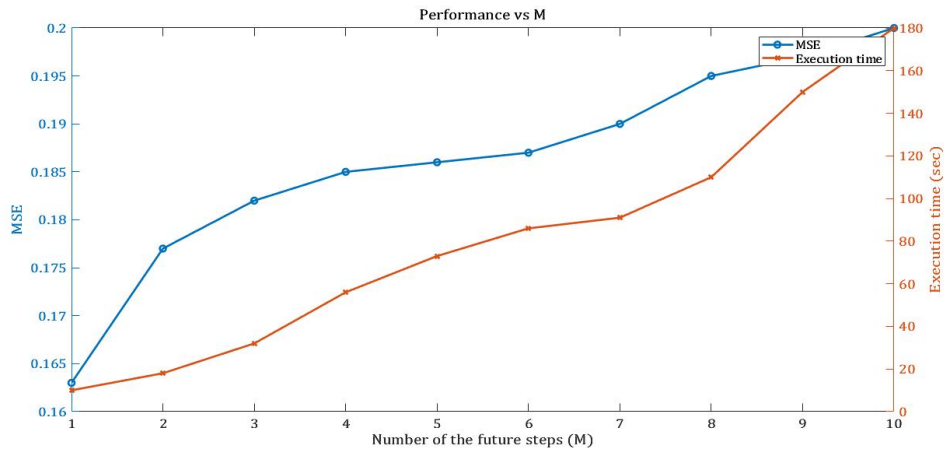


Figure 5.5: The relation among MSE, Execution time, and M

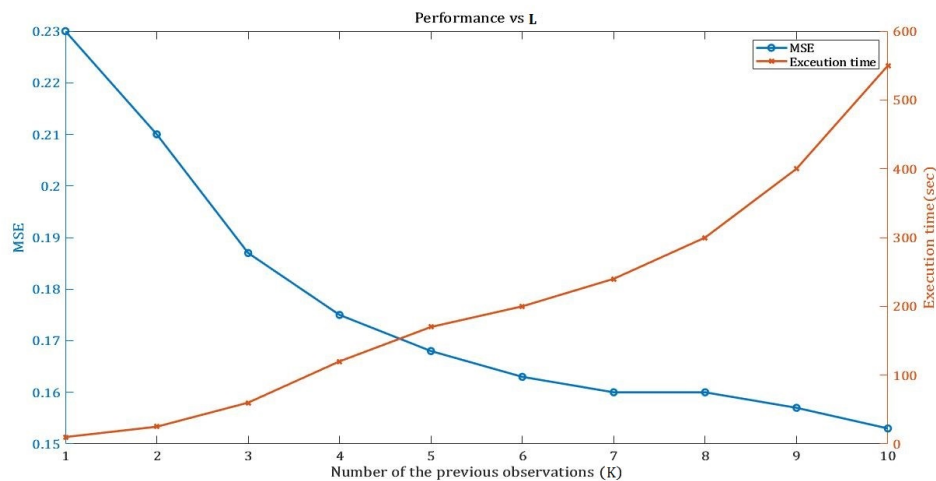


Figure 5.6: The relation among MSE, Execution time, and K

5.7 Conclusions

In this thesis, I used the FFNN machine learning model to build a PdM system for WSN; it predicts the operational status (either being "OK" or faulty) after M time steps based on K previous readings of QoS requirements of the WSN. I used real estate data set of one-hop WSN. I also used quantization and encoding schemes to make the system incoherent with the limited resources of the WSN. I revealed that the complexity of systems is improved by quantization, encoding, small M and small K . The accuracy is improved by using the raw(original data), small M , and large k .

Chapter 6

Summary of results and thesis

6.1 Summary of results

As wireless data acquisition tends to become the fundamental means of robust industrial control and digitization, this dissertation has proposed some innovative techniques for designing robust WSNs for optimal energy efficiency, reliability, and longevity.

I introduced new algorithm, such as MMREP (Maximum of the Minimum Residual Energy Protocol) and OREBP (Optimal Residual Energy Based Protocol) to maximize the minimum remaining energy and to achieve uniform energy consumption subject to a predefined reliability constraint. The proposed protocols improve energy efficiency in terms of the life span as they minimize the amount, of consumed energy and distribute the load evenly.

I have also proposed a new algorithm, BPEEBP (BackPressure Energy-Entropy-based Protocol), for stochastic WSNs. It controls the distribution of the traffic and the residual energy subject to a predefined reliability constraint. It exhibits better performance regarding energy efficiency, E2E delay, and throughput. The proposed algorithms can be used in WSNs and any routing-based systems such as telecommunication and transportation; they can also be used in systems that need load balance, such as resources management systems.

I developed a new optimization method called CCS-MOGA (Chained Compressive Sensing based on Multiple Objects Genetic Algorithm), to find the optimal values of the variables of the compressive sensing paradigm. I optimized the number of measurements, transmission range, and the mutual coherence of the sensing matrix. CCS-MOGA maximizes the energy efficiency and minimizes the probability of reconstruction error. It also provides a dynamic construction of WSN based on the values of optimization variables and objectives. In addition to WSN, the developed algorithm can be used as a heuristic optimization algorithm to strike the right balance between contradicting criteria of networking. The new method can be used in the structure of the graph-based application and optimal resource distribution map.

I used the FFNN machine learning model to build a PdM system for WSN;

it predicts the operational status (“OK” or faulty) after M time steps based on L previous readings of QoS requirements of the WSN. I used the real estate data set of a one-hop WSN. I also used quantization and encoding schemes to make the system incoherent with the limited resources of the WSN. I revealed that the complexity of systems is improved by quantization, encoding, small M , and small L . The accuracy is improved by using the raw(original data), small M , and large k .

In this way, my research has contributed to industrial and IoT data acquisition, which may help intelligent system monitoring. The proposed algorithms can be used as efficient data collection for AI-based system supervision. They had several applications ranging from surveillance systems to Industry 4.0.

6.2 The novel scientific results of the dissertation

Thesis 1:

- I presented a method for selecting the relay nodes based on the remaining energy and reliability.
- I proposed MMREP to select the path that maximizes the remaining energy of its nodes subject to the reliability constraint.
- I introduced the entropy-like function to measure the uniformity of residual energy distribution.
- I proposed OREBP, which selects the path based on the entropy of residual energy of the path.

Thesis 2:

- I proposed BPEEBP, a routing algorithm for stochastic WSNs; it based on four significant principles; On-OFF scheme to provide energy-efficiency, backpressure principle to control the traffic load balance, entropy of residual energy to control the uniform distribution of residual energy, and Rayleigh fading model is to guarantee the balance between consumed energy and reliability.

Thesis 3:

- I proposed a multiple objectives genetic algorithm to strike the right balance between the energy efficiency and reconstruction error of the compressive sensing method.
- I applied a genetic algorithm with appropriate parameters to improve the characteristics of the sensing matrix by reducing its mutual coherence.

- I presented a greedy algorithm to split the WSN into multiple paths based on the above optimization algorithms.

Thesis 4:

- I proposed a prediction model that estimates the forward probability distribution of the operational status of the monitored system in the next M future steps based on previous K readings of QoS parameters.
- I implemented the proposed prediction model by FFNN model.
- I presented a mathematical analysis for configured, measured, and calculated QoS parameters of WSNs.
- I applied the proposed model as PdMs for WSN; it predicts the needing of WSN for maintenance after M steps, the prediction is based on previous K observations of QoS requirements.
- I introduced quantization and encoding schemes to make the proposed model compatible with limited resource WSNs.

6.3 Possible future works

I hope my dissertation may contribute to opening new horizons in the WSNs and IoT networks. I also hope that other researchers and I can extend the proposed algorithms into new dimensions of performance and applications.

In the energy-efficient routing algorithms, there is a need to develop new routing algorithms that hit a good balance between energy efficiency and complexity; the area still needs mathematical optimization algorithms to find the optimal lifespan, data loss percentage, the maximum number of hops, and complexity. In stochastic WSN, the need still exists to extend the proposed algorithm to include multi-hops routing and assume asymmetrical nodes concerning packet generation rate, service rates, and buffer size.

In the area of data reduction based on compressive sensing, the proposed algorithm is based on a heuristic optimization method to strike a good-enough balance between reconstruction error (as a proxy for reliability) and energy efficiency. More research is needed to find mathematically the optimal solution that strikes the right balance among the parameters of compressive sensing and the QoS parameters of WSNs. Exploring other application areas of the proposed algorithm also is a promising domain.

For as PdMs, the proposed model can be extended in two primary dimensions: first, application of the proposed estimation model in other fields such as industry, agriculture, transportation..., etc. Second, implementation of the proposed model by different machine learning and deep learning techniques. Other techniques may improve the performance in terms of accuracy, complexity, and ease of implementation.

In general, the proposed algorithms can be used as efficient data collection for AI-based system supervision. They can be modified and adapted for use in other applications ranging from surveillance systems to Industry 4.0.

Publications

1. Almazaidh, M., & Levendovszky, J. (2020). Novel Reliable and Energy-Efficient Routing Protocols for Wireless Sensor Networks. *Journal of Sensor and Actuator Networks*, 9(1), 5. <https://doi.org/10.3390/jsan9010005>
2. Almazaidh, M., & Levendovszky, J. (2020a). A New Energy-Efficient and Reliable Protocol for Stochastic WSNs Based on Back-Pressure and Entropy of Residual Energy. *Journal of Computer and Communications*, 08(11), 1–13. <https://doi.org/10.4236/jcc.2020.811001>
3. Mazaideh, M. A., & Levendovszky, J. (2021). A multi-hop routing algorithm for WSNs based on compressive sensing and multiple objective genetic algorithm. *Journal of Communications and Networks*, 23(2), 138–147. <https://doi.org/10.23919/jcn.2021.000003>
4. Mohammed Almazaidh and Janos Levendovszky. Predictive maintenance system for WSNs: A machine learning approach. *Indonesian Journal of Electrical Engineering and Computer Science*, (p-ISSN: 2502-4752, e-ISSN: 2502-4760). Accepted

Bibliography

- [1] Mohammed Almazaideh and Janos Levendovszky. Novel reliable and energy-efficient routing protocols for wireless sensor networks. *Journal of Sensor and Actuator Networks*, 9(1):5, 2020.
- [2] Mohammed Almazaideh and Janos Levendovszky. A new energy-efficient and reliable protocol for stochastic wsns based on back-pressure and entropy of residual energy. *Journal of Computer and Communications*, 8(11):1–13, 2020.
- [3] Mohammed Al Mazaideh and Janos Levendovszky. A multi-hop routing algorithm for wsns based on compressive sensing and multiple objective genetic algorithm. *Journal of Communications and Networks*, (99):1–10, 2021.
- [4] Ian F Akyildiz and Mehmet Can Vuran. *Wireless sensor networks*, volume 4. John Wiley & Sons, 2010.
- [5] Anis Koubâa, Mário Alves, and Eduardo Tovar. Ieee 802.15. 4 for wireless sensor networks: a technical overview. 2005.
- [6] Bhaskar Bhuyan, Hiren Kumar Deva Sarma, Nityananda Sarma, Avijit Kar, Rajib Mall, et al. Quality of service (qos) provisions in wireless sensor networks and related challenges. *Wireless Sensor Network*, 2(11):861, 2010.
- [7] Ridha Soua and Pascale Minet. A survey on energy efficient techniques in wireless sensor networks. In *2011 4th Joint IFIP Wireless and Mobile Networking Conference (WMNC 2011)*, pages 1–9. IEEE, 2011.
- [8] Tayyaba Liaqat, Mariam Akbar, Nadeem Javaid, Umar Qasim, Zahoor Ali Khan, Qaisar Javaid, Turki Ali Alghamdi, and Iftikhar Azim Niaz. On reliable and efficient data gathering based routing in underwater wireless sensor networks. *Sensors*, 16(9):1391, 2016.
- [9] Zahra Taghikhaki, Nirvana Meratnia, and Paul JM Havinga. A reliable and energy-efficient chain-cluster based routing protocol for wireless sensor networks. In *2013 IEEE eighth international conference on intelligent sensors, sensor networks and information processing*, pages 248–253. IEEE, 2013.

-
- [10] Zheng-Yang Ai, Yu-Tong Zhou, and Fei Song. A smart collaborative routing protocol for reliable data diffusion in iot scenarios. *Sensors*, 18(6):1926, 2018.
- [11] Scott Moeller, Avinash Sridharan, Bhaskar Krishnamachari, and Omprakash Gnawali. Routing without routes: The backpressure collection protocol. In *Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks*, pages 279–290, 2010.
- [12] Eleftheria Athanasopoulou, Loc X Bui, Tianxiong Ji, R Srikant, and Alexander Stolyar. Back-pressure-based packet-by-packet adaptive routing in communication networks. *IEEE/ACM transactions on networking*, 21(1):244–257, 2012.
- [13] Zhenzhen Jiao, Rui Tian, Baoxian Zhang, and Cheng Li. A gradient-assisted energy-efficient backpressure scheduling algorithm for wireless sensor networks. *International Journal of Distributed Sensor Networks*, 11(11):460506, 2015.
- [14] Fawaz Alassery. A new link weight factor in backpressure scheduling algorithm for energy-efficient design of smart wireless sensor networks. In *2016 IEEE Smart Energy Grid Engineering (SEGE)*, pages 365–370. IEEE, 2016.
- [15] Meenu Rani, Sanjay B Dhok, and Raghavendra B Deshmukh. A systematic review of compressive sensing: Concepts, implementations and applications. *IEEE Access*, 6:4875–4894, 2018.
- [16] David L Donoho. Compressed sensing. *IEEE Transactions on information theory*, 52(4):1289–1306, 2006.
- [17] Irena Orović, Vladan Papić, Cornel Ioana, Xiumei Li, and Srdjan Stanković. Compressive sensing in signal processing: algorithms and transform domain formulations. *Mathematical Problems in Engineering*, 2016, 2016.
- [18] Rashmi Ranjan Rout and Soumya K Ghosh. Adaptive data aggregation and energy efficiency using network coding in a clustered wireless sensor network: An analytical approach. *Computer Communications*, 40:65–75, 2014.
- [19] Yongfa Hong, Juan Xu, and Changjun Jiang. Lifetime maximization in wireless sensor networks with network coding. In *2007 International Conference on Wireless Communications, Networking and Mobile Computing*, pages 2527–2530. IEEE, 2007.
- [20] Nildo dos Santos Ribeiro Júnior, Rodrigo C Tavares, Marcos AM Vieira, Luiz FM Vieira, and Omprakash Gnawali. Codedrip: Improving data

- dissemination for wireless sensor networks with network coding. *Ad Hoc Networks*, 54:42–52, 2017.
- [21] Rajarshi Middy, Nabajit Chakravarty, and Mrinal Kanti Naskar. Compressive sensing in wireless sensor networks—a survey. *IETE technical review*, 34(6):642–654, 2017.
- [22] Ahmed Aziz, Karan Singh, Walid Osamy, and Ahmed M Khedr. Effective algorithm for optimizing compressive sensing in iot and periodic monitoring applications. *Journal of Network and Computer Applications*, 126:12–28, 2019.
- [23] Yongjun Zhu, Wenbo Liu, and Qian Shen. Adaptive algorithm on block-compressive sensing and noisy data estimation. *Electronics*, 8(7):753, 2019.
- [24] Salvatore T March and Gary D Scudder. Predictive maintenance: strategic use of it in manufacturing organizations. *Information Systems Frontiers*, 21(2):327–341, 2019.
- [25] Federico Civerchia, Stefano Bocchino, Claudio Salvadori, Enrico Rossi, Luca Maggiani, and Matteo Petracca. Industrial internet of things monitoring solution for advanced predictive maintenance applications. *Journal of Industrial Information Integration*, 7:4–12, 2017.
- [26] Donato Catenazzo, Brendan O’Flynn, and Michael Walsh. On the use of wireless sensor networks in preventative maintenance for industry 4.0. In *2018 12th International Conference on Sensing Technology (ICST)*, pages 256–262. IEEE, 2018.
- [27] Ning Cao, Yongbin Zhao, Jing Liang, Tiantian Wang, Teng Huang, Dongchen Xu, and Yuzhen Xu. The comparison of single-hop and leach protocols in wireless sensor networks. In *2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC)*, volume 2, pages 426–429. IEEE, 2017.
- [28] Mohamed Amine Kafi, Jalel Ben Othman, and Nadjib Badache. A survey on reliability protocols in wireless sensor networks. *ACM Computing Surveys (CSUR)*, 50(2):1–47, 2017.
- [29] Tifenn Rault, Abdelmadjid Bouabdallah, and Yacine Challal. Energy efficiency in wireless sensor networks: A top-down survey. *Computer Networks*, 67:104–122, 2014.
- [30] Junaid Ahmed Khan, Hassaan Khaliq Qureshi, and Adnan Iqbal. Energy management in wireless sensor networks: A survey. *Computers and Electrical Engineering*, 41:159–176, 2015.

- [31] Trakadas P Zahariadis T. *Design Guidelines for Routing Metrics Composition in LLN*.
- [32] Reem E Mohamed, Walid R Ghanem, Abeer T Khalil, Mohamed El-hoseny, Muhammad Sajjad, and Mohamed A Mohamed. Energy efficient collaborative proactive routing protocol for wireless sensor network. *Computer Networks*, 142:154–167, 2018.
- [33] Gergely Treplan, Long Tran-Thanh, and Janos Levendovszky. Energy efficient reliable cooperative multipath routing in wireless sensor networks. *World Academy of Science, Engineering and Technology*, 68:1366–1371, 2010.
- [34] Wendi Rabiner Heinzelman, Anantha Chandrakasan, and Hari Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. In *Proceedings of the 33rd annual Hawaii international conference on system sciences*, pages 10–pp. IEEE, 2000.
- [35] Nikolaos A Pantazis, Stefanos A Nikolidakis, and Dimitrios D Vergados. Energy-efficient routing protocols in wireless sensor networks: A survey. *IEEE Communications surveys & tutorials*, 15(2):551–591, 2012.
- [36] Shweta Varshney and Rakesh Kuma. Variants of leach routing protocol in wsn: A comparative analysis. In *2018 8th International conference on cloud computing, data science & engineering (confluence)*, pages 199–204. IEEE, 2018.
- [37] Wang Ke, Ou Yangrui, Ji Hong, Zhang Heli, and Li Xi. Energy aware hierarchical cluster-based routing protocol for wsns. *The Journal of China Universities of Posts and Telecommunications*, 23(4):46–52, 2016.
- [38] Qingjian Ni, Qianqian Pan, Huimin Du, Cen Cao, and Yuqing Zhai. A novel cluster head selection algorithm based on fuzzy clustering and particle swarm optimization. *IEEE/ACM transactions on computational biology and bioinformatics*, 14(1):76–84, 2015.
- [39] Amit Sarkar and T Senthil Murugan. Cluster head selection for energy efficient and delay-less routing in wireless sensor network. *Wireless Networks*, 25(1):303–320, 2019.
- [40] Boseon Yu, Wonik Choi, Taikjin Lee, and Hyunduk Kim. Clustering algorithm considering sensor node distribution in wireless sensor networks. *Journal of Information Processing Systems*, 14(4):926–940, 2018.
- [41] Lakshmanarao Battula and P Vamsikrishna Raja. Power efficient gathering in sensor information systems protocol using k-means clustering algorithm. *International Journal of Science, Engineering and Computer Technology*, 6(4):133, 2016.

- [42] Kuong-Ho Chen, Jyh-Ming Huang, and Chieh-Chuan Hsiao. Chiron: an energy-efficient chain-based hierarchical routing protocol in wireless sensor networks. In *2009 Wireless Telecommunications Symposium*, pages 1–5. IEEE, 2009.
- [43] Yongchang Yu and Yichang Song. An energy-efficient chain-based routing protocol in wireless sensor network. In *2010 International conference on computer application and system modeling (ICCASM 2010)*, volume 11, pages V11–486. IEEE, 2010.
- [44] Alekha Kumar Mishra, Rukshan Ur Rahman, Rahul Bharadwaj, and Rohit Sharma. An enhancement of pegasis protocol with improved network lifetime for wireless sensor networks. In *2015 IEEE Power, Communication and Information Technology Conference (PCITC)*, pages 142–147. IEEE, 2015.
- [45] Natasha Ramluckun and Vandana Bassoo. Energy-efficient chain-cluster based intelligent routing technique for wireless sensor networks. *Applied Computing and Informatics*, 2020.
- [46] Lorenzo Keller, Emre Atsan, Katerina Argyraki, and Christina Fragouli. Sensecode: Network coding for reliable sensor networks. *ACM Transactions on Sensor Networks (TOSN)*, 9(2):1–20, 2013.
- [47] JÁNOS Levendovszky, A Bojársky, B Karlocai, and A Olah. Energy balancing by combinatorial optimization for wireless sensor networks. *WSEAS Transactions on Communications*, 7(2):27–32, 2008.
- [48] Shruti Lall, Attahiru Sule Alfa, and Bodhaswar T Maharaj. The role of queueing theory in the design and analysis of wireless sensor networks: An insight. In *2016 IEEE 14th International Conference on Industrial Informatics (INDIN)*, pages 1191–1194. IEEE, 2016.
- [49] Tie Qiu, Feng Xia, Lin Feng, Guowei Wu, and Bo Jin. Queueing theory-based path delay analysis of wireless sensor networks. *Advances in Electrical and Computer Engineering*, 11(2):3–8, 2011.
- [50] Lulu Liang, Deyun Gao, and Victor CM Leung. Queue-based congestion detection and multistage rate control in event-driven wireless sensor networks. *Wireless Communications and Mobile Computing*, 14(8):818–830, 2014.
- [51] Heejung Byun and Junglok Yu. Adaptive duty cycle control with queue management in wireless sensor networks. *IEEE Transactions on Mobile Computing*, 12(6):1214–1224, 2012.
- [52] Marco Martalò, Stefano Busanelli, and Gianluigi Ferrari. Markov chain-based performance analysis of multihop ieee 802.15. 4 wireless networks. *Performance Evaluation*, 66(12):722–741, 2009.

- [53] Leandros Tassiulas and Anthony Ephremides. Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks. In *29th IEEE Conference on Decision and Control*, pages 2130–2132. IEEE, 1990.
- [54] Zhenzhen Jiao, Baoxian Zhang, Cheng Li, and Hussein T Mouftah. Backpressure-based routing and scheduling protocols for wireless multihop networks: A survey. *IEEE Wireless Communications*, 23(1):102–110, 2016.
- [55] Lei Ying, Sanjay Shakkottai, Aneesh Reddy, and Shihuan Liu. On combining shortest-path and back-pressure routing over multihop wireless networks. *IEEE/ACM Transactions on Networking*, 19(3):841–854, 2010.
- [56] Natale Patriciello, José Núñez-Martínez, Jorge Baranda, Maurizio Casoni, and Josep Mangués-Bafalluy. Towards backpressure routing in wireless mesh backhauls for dense lte deployments. In *2019 International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, pages 282–287. IEEE, 2019.
- [57] Yad Tahir, Shusen Yang, and Julie McCann. Brpl: Backpressure rpl for high-throughput and mobile iots. *IEEE Transactions on Mobile Computing*, 17(1):29–43, 2017.
- [58] Rong Zheng, Jennifer C Hou, and Lui Sha. Asynchronous wakeup for ad hoc networks. In *Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing*, pages 35–45, 2003.
- [59] Harsh Bhatia, RB Lenin, Aarti Munjal, Srini Ramaswamy, and Sanjay Srivastava. A queuing-theoretic framework for modeling and analysis of mobility in wsns. In *Proceedings of the 8th Workshop on Performance Metrics for Intelligent Systems*, pages 248–253, 2008.
- [60] Mustafa K Mehmet Ali and Hao Gu. A performance modeling of wireless sensor networks as a queueing network with on and off servers. *Journal of Communications and Networks*, 11(4):406–415, 2009.
- [61] Emmanuel J Candès, Justin Romberg, and Terence Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on information theory*, 52(2):489–509, 2006.
- [62] Emmanuel J Candes and Terence Tao. Near-optimal signal recovery from random projections: Universal encoding strategies? *IEEE transactions on information theory*, 52(12):5406–5425, 2006.
- [63] LiangHui Ding, Ping Wu, Hao Wang, ZhiWen Pan, and XiaoHu You. Lifetime maximization routing with network coding in wireless multihop networks. *Science China Information Sciences*, 56(2):1–15, 2013.

- [64] Hisham Alshaheen and Haifa Takruri Rizk. Improving the energy efficiency for biosensor nodes in the wbsn bottleneck zone based on a random linear network coding. In *2017 11th International Symposium on Medical Information and Communication Technology (ISMICT)*, pages 59–63. IEEE, 2017.
- [65] L. Xiang, J. Luo, and A. Vasilakos. Compressed data aggregation for energy efficient wireless sensor networks. In *2011 8th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, pages 46–54, 2011.
- [66] Cuicui Lv, Qiang Wang, Wenjie Yan, and Yi Shen. Energy-balanced compressive data gathering in wireless sensor networks. *Journal of Network and Computer Applications*, 61:102–114, 2016.
- [67] Minh T Nguyen and Keith A Teague. Compressive sensing based random walk routing in wireless sensor networks. *Ad Hoc Networks*, 54:99–110, 2017.
- [68] Ahmed Salim and Walid Osamy. Distributed multi chain compressive sensing based routing algorithm for wireless sensor networks. *Wireless Networks*, 21(4):1379–1390, 2015.
- [69] Quan Wang, Deyu Lin, Pengfei Yang, and Zhiqiang Zhang. An energy-efficient compressive sensing-based clustering routing protocol for wsns. *IEEE Sensors Journal*, 19(10):3950–3960, 2019.
- [70] Siguang Chen, Chuanxin Zhao, Meng Wu, Zhixin Sun, Haijun Zhang, and Victor CM Leung. Compressive network coding for wireless sensor networks: Spatio-temporal coding and optimization design. *Computer Networks*, 108:345–356, 2016.
- [71] Emmanuel J Candès and Michael B Wakin. An introduction to compressive sampling. *IEEE signal processing magazine*, 25(2):21–30, 2008.
- [72] Zesong Fei, Bin Li, Shaoshi Yang, Chengwen Xing, Hongbin Chen, and Lajos Hanzo. A survey of multi-objective optimization in wireless sensor networks: Metrics, algorithms, and open problems. *IEEE Communications Surveys & Tutorials*, 19(1):550–586, 2016.
- [73] Muhammad Iqbal, Muhammad Naeem, Alagan Anpalagan, Nadia N Qadri, and M Imran. Multi-objective optimization in sensor networks: Optimization classification, applications and solution approaches. *Computer Networks*, 99:134–161, 2016.
- [74] Ujjwal Maulik, Sanghamitra Bandyopadhyay, and Anirban Mukhopadhyay. Genetic algorithms and multiobjective optimization. In *Multiobjective genetic algorithms for clustering*, pages 25–50. Springer, 2011.

- [75] Ji Li, Lachlan LH Andrew, Chuan Heng Foh, Moshe Zukerman, and Hsiao-Hwa Chen. Connectivity, coverage and placement in wireless sensor networks. *Sensors*, 9(10):7664–7693, 2009.
- [76] Haojun Teng, Kuan Zhang, Mianxiong Dong, Kaoru Ota, Anfeng Liu, Ming Zhao, and Tian Wang. Adaptive transmission range based topology control scheme for fast and reliable data collection. *Wireless Communications and Mobile Computing*, 2018, 2018.
- [77] Ce Zhang, Xia Zhang, Ou Li, Yanping Yang, and Guangyi Liu. Dynamic clustering and compressive data gathering algorithm for energy-efficient wireless sensor networks. *International Journal of Distributed Sensor Networks*, 13(10):1550147717738905, 2017.
- [78] Yongyi Ran, Xin Zhou, Pengfeng Lin, Yonggang Wen, and Ruilong Deng. A survey of predictive maintenance: Systems, purposes and approaches. *arXiv preprint arXiv:1912.07383*, 2019.
- [79] Rolf Isermann. *Fault-diagnosis systems: an introduction from fault detection to fault tolerance*. Springer Science & Business Media, 2005.
- [80] Oscar Serradilla, Ekhi Zugasti, and Urko Zurutuza. Deep learning models for predictive maintenance: a survey, comparison, challenges and prospect. *arXiv preprint arXiv:2010.03207*, 2020.
- [81] Thyago P Carvalho, Fabrizzio AAMN Soares, Roberto Vita, Roberto da P Francisco, João P Basto, and Symone GS Alcalá. A systematic literature review of machine learning methods applied to predictive maintenance. *Computers & Industrial Engineering*, 137:106024, 2019.
- [82] Ashok Prajapati, James Bechtel, and Subramaniam Ganesan. Condition based maintenance: a survey. *Journal of Quality in Maintenance Engineering*, 2012.
- [83] Hyoung Seok Kang, Ju Yeon Lee, SangSu Choi, Hyun Kim, Jun Hee Park, Ji Yeon Son, Bo Hyun Kim, and Sang Do Noh. Smart manufacturing: Past research, present findings, and future directions. *International journal of precision engineering and manufacturing-green technology*, 3(1):111–128, 2016.
- [84] Sule Selcuk. Predictive maintenance, its implementation and latest trends. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 231(9):1670–1679, 2017.
- [85] Behzad V Farahani, Francisco Barros, Pedro J Sousa, Paulo J Tavares, and Pedro MGP Moreira. A railway tunnel structural monitoring methodology proposal for predictive maintenance. *Structural Control and Health Monitoring*, 27(8):e2587, 2020.

- [86] Benoîte de Saporta, François Dufour, Huilong Zhang, and Charles Elegbede. Optimal stopping for the predictive maintenance of a structure subject to corrosion. *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, 226(2):169–181, 2012.
- [87] Alessandro Massaro, Angelo Galiano, Giacomo Meuli, and Saverio Francesco Massari. Overview and application of enabling technologies oriented on energy routing monitoring, on network installation and on predictive maintenance. *International Journal of Artificial Intelligence and Applications (IJAIA)*, 9(2), 2018.
- [88] Irfan Ullah, Fan Yang, Rehanullah Khan, Ling Liu, Haisheng Yang, Bing Gao, and Kai Sun. Predictive maintenance of power substation equipment by infrared thermography using a machine-learning approach. *Energies*, 10(12):1987, 2017.
- [89] Phattara Khumprom, David Grewell, and Nita Yodo. Deep neural network feature selection approaches for data-driven prognostic model of aircraft engines. *Aerospace*, 7(9):132, 2020.
- [90] Soonsung Hwang, Jongpil Jeong, and Youngbin Kang. Svm-rbm based predictive maintenance scheme for iot-enabled smart factory. In *2018 Thirteenth International Conference on Digital Information Management (ICDIM)*, pages 162–167. IEEE, 2018.
- [91] Ruonan Liu, Guotao Meng, Boyuan Yang, Chuang Sun, and Xuefeng Chen. Dislocated time series convolutional neural architecture: An intelligent fault diagnosis approach for electric machine. *IEEE Transactions on Industrial Informatics*, 13(3):1310–1320, 2016.
- [92] Mohammed Ali Al-Garadi, Amr Mohamed, Abdulla Khalid Al-Ali, Xiaojiang Du, Ihsan Ali, and Mohsen Guizani. A survey of machine and deep learning methods for internet of things (iot) security. *IEEE Communications Surveys & Tutorials*, 22(3):1646–1685, 2020.
- [93] Tao Liu and Alberto E Cerpa. Data-driven link quality prediction using link features. *ACM Transactions on Sensor Networks (TOSN)*, 10(2):1–35, 2014.
- [94] Merima Kulin, Eli De Poorter, Tarik Kazaz, and Ingrid Moerman. Poster: Towards a cognitive mac layer: Predicting the mac-level performance in dynamic wsn using machine learning. In *Proceedings of the 2017 International Conference on Embedded Wireless Systems and Networks*, pages 214–215, 2017.
- [95] Ayhan Akbas, Huseyin Ugur Yildiz, Ahmet Murat Ozbayoglu, and Bulent Tavli. Neural network based instant parameter prediction for wireless sensor network optimization models. *Wireless Networks*, 25(6):3405–3418, 2019.

- [96] Muhammad Ateeq, Farruh Ishmanov, Muhammad Khalil Afzal, and Muhammad Naeem. Multi-parametric analysis of reliability and energy consumption in iot: A deep learning approach. *Sensors*, 19(2):309, 2019.
- [97] Attila Ceffer, Norbert Fogarasi, and Janos Levendovszky. Trading by estimating the quantized forward distribution. *Applied Economics*, 50(59):6397–6405, 2018.
- [98] Christos Louizos, Max Welling, and Diederik P Kingma. Learning sparse neural networks through l_0 regularization. *arXiv preprint arXiv:1712.01312*, 2017.
- [99] Antonino Laudani, Gabriele Maria Lozito, Francesco Riganti Fulginei, and Alessandro Salvini. On training efficiency and computational costs of a feed forward neural network: a review. *Computational intelligence and neuroscience*, 2015, 2015.
- [100] Benoit Jacob, Skirmantas Kligys, Bo Chen, Menglong Zhu, Matthew Tang, Andrew Howard, Hartwig Adam, and Dmitry Kalenichenko. Quantization and training of neural networks for efficient integer-arithmetic-only inference. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2704–2713, 2018.
- [101] Hao Li, Soham De, Zheng Xu, Christoph Studer, Hanan Samet, and Tom Goldstein. Training quantized nets: A deeper understanding. *arXiv preprint arXiv:1706.02379*, 2017.
- [102] Nimit Sharad Sohoni, Christopher Richard Aberger, Megan Leszczynski, Jian Zhang, and Christopher Ré. Low-memory neural network training: A technical report. *arXiv preprint arXiv:1904.10631*, 2019.
- [103] David Kotz, Tristan Henderson, Ilya Abyzov, and Jihwang Yeo. CRAW-DAD dataset dartmouth/campus (v. 2009-09-09). Downloaded from <https://crawdad.org/dartmouth/campus/20090909>, September 2009.
- [104] Songwei Fu, Yan Zhang, Yuming Jiang, Chengchen Hu, Chia-Yen Shih, and Pedro José Marrón. Experimental study for multi-layer parameter configuration of wsn links. In *2015 IEEE 35th International Conference on Distributed Computing Systems*, pages 369–378. IEEE, 2015.
- [105] Ivana Tomić and Julie A McCann. A survey of potential security issues in existing wireless sensor network protocols. *IEEE Internet of Things Journal*, 4(6):1910–1923, 2017.