AN ABSTRACT OF THE DISSERTATION OF

Yousef Qassim for the degree of Doctor of Philosophy in Electrical and Computer Engineering presented on December 17, 2019.

Title: Coding Techniques for Achieving Efficient Wireless Sensor Networks

Abstract approved: _____

Mario Magaña

In this dissertation, we explored multiple coding techniques to reduce energy consumption, improve performance, and secure wireless sensor networks specifically and ad-hoc networks in general. With the introduction of Internet of Things (IoT) and 5G technologies, wireless sensor networks are quickly emerging as an important and key technology in the future. From their ability to sense, process, and communicate data among them to being low-powered, self organizing, and cost effective. Their characteristics made them a great tool for many applications, they already have a role in connecting homes, cars, surveillance systems, early earthquake and forest fire detection. However, due to their limited power and processing energy, they suffer to maintain acceptable performance and connectivity especially when deployed in harsh environment. In this research, we demonstrated novel techniques that can help improve their performance while reducing energy consumption. The contribution of this work is summarized below.

- We propose a novel approach to error correction codes in wireless sensor network. We introduce a modification to Reed-Solomon decoding algorithm which allows errors to occur in data without sacrificing the total integrity of the data. We show that by deploying such mechanisms, we can reduce the total energy required to deliver data at their destination by reducing the decoding energy per symbol/bit.
- We propose a modification on opportunistic network coding (ONC) using diversity coding and cooperation, as well as, limiting the number of packets that can be network-coded together to three and only encode packets that were received by relay nodes directly. We show that using such techniques we can alleviate the issues that plague ONC when implemented in noisy networks. We study the effect of link outages/mobility on proposed solution and show that our proposed solution can accommodate up to one link failure.

- We study the security of ad-hoc networks and propose a post-quantum hybrid security mechanism. We propose a security mechanism that take advantage of the wireless medium hereditary nature and cryptography techniques. This state of art protocol is able to overcome the presence of adversary eavesdropper and address man in the middle attack. Our security mechanism uses a combination of physical layer and cryptographic security techniques to provide best effort security.

Coding Techniques for Achieving Efficient Wireless Sensor Networks

by

Yousef Qassim

A DISSERTATION

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Doctor of Philosophy

Presented December 17, 2019
Commencement June 2020

Doctor of Philosophy dissertation of Yousef Qassim presented on December 17, 2019.

APPROVED:

Major Professor, representing Electrical and Computer Engineering

---

Head of the School of Electrical Engineering and Computer Science

---

Dean of the Graduate School

I understand that my dissertation will become part of the permanent collection of Oregon State University libraries. My signature below authorizes release of my dissertation to any reader upon request.

Yousef Qassim, Author

ACKNOWLEDGEMENTS

"He who does not thank Allah (God), does not thank people"

-Muhammad Messenger of Allah.

I start by showing my deep appreciation to Allah the most merciful, the most gracious, for bestowing upon me his blessings and giving me the strength to fulfill my journey toward Ph.D. I like to offer my gratitude to my parents, Montaser and Eman for supporting me all these years and raising me to be the man I aspire. I like to thank my grandmother for being my first and biggest fan. Also, I would like to thank my wife Tamara who stood all those years beside me and provided me with her undivided attention who without I would not accomplished my goals. I like to thank my daughter and son, Waard and Montaser for being the fruits of my life and making my journey worthwhile. I like to thank my brothers, Hani, Rida, Mohammad, and Jawad for believing in me and being there when I needed them.

I like to thank Prof. Mario Magaña for his support, encouragement, and guidance. He believed in me even when I didn't believe in myself. I'm honored to call him a mentor and a friend. I would like to give my sincere appreciation to Prof. Bella Bose, Prof. Bechir Hamdaoui, Prof. Yue Cao, and Prof. William Warnes for acting as my committee and providing me with their valuable feedback. I also like to thank Prof. Haupping Luo and Prof. Ben Lee for being part of my qualification and preliminary exam committees. I like to thank prof. Attila Yavuz for his valuable feedback on our security paper.

I like to thank Dr. Jon Dorbolo for providing me with a job on campus and for being the person he is. I like to thank Kelley engineering center staff, my colleagues, and friends for their continuous support and making my time at Oregon State University a pleasant one.

To all of you thank you from the bottom of my heart, really without you my journey would not reach an end.

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

LIST OF FIGURES

LIST OF FIGURES (Continued)

LIST OF TABLES

LIST OF ALGORITHMS

For my wife, daughter, and son
To my life among you
With love...

Chapter 1: Introduction

Wireless sensor networks have been envisioned as one of most important technologies. The emerging advancements in the field of microelectronics and wireless communications have allowed the development and deployment of wireless sensor networks. Wireless sensor networks consist of many sensor nodes that range from hundreds to thousands depending on the phenomena being observed within an environment. These low-power small units are able to sense, process, and communicate with each other. They are widely deployed in different fields such as military, environment, health, and industry. Their applications range from battle field surveillance, to early earth quake and forest fire detection, to factories and highways monitoring [1]. An example of wireless sensor networks depicted in Figure 1.1.

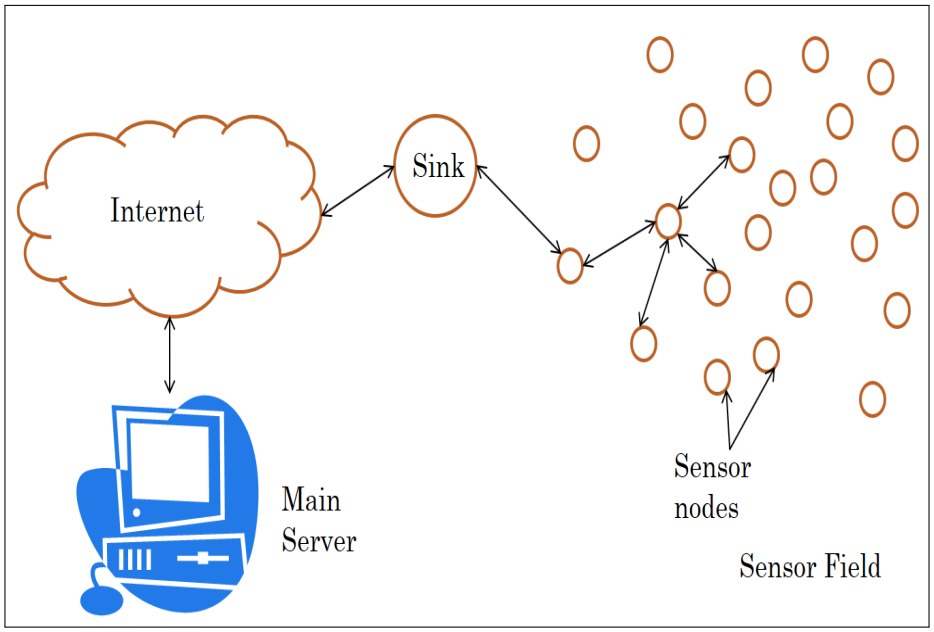

Figure 1.1: Wireless sensor network.

To accomplish such applications, wireless sensor networks must have the ability of self organization, cooperative data processing, and cooperative communication between their nodes. Due to sensor node failures, wireless sensor networks cannot always be engineered and designed. Therefore wireless sensor networks require having random topologies particularly in remote areas where accessibility is impossible. This demands the need of networks with ad-hoc nature. The differences between traditional ad-hoc networks and wireless sensor networks are summarized in [1] as follows:

- The number of nodes in wireless sensor networks is much higher than the number of nodes in ad-hoc networks.
- Sensor nodes are subject to failure and they are deployed in large numbers.
- The topology of wireless sensor network changes frequently due to node failures or movement.
- Wireless sensor networks are limited in their capabilities, e.g. power, processing, communication, and memory.
- Wireless sensor networks sometimes have to extract data to compute certain function such as average, while traditional ad-hoc network are only concerned with end-to-end communication.
- Sensor nodes might lack global identification due to their large numbers.

In wireless sensor networks usually the goal is not to collect raw data about the phenomena being observed, but to compute an aggregate function of these data. Data aggregation, also known as data fusion, is a method used to combine the measured data in a meaningful way. Aggregation functions such as computing the average, the maximum, or the minimum help in reducing the amount of data transmitted over the network and increase the network reliability.

Sensor nodes are usually battery operated and deployed in harsh and unreachable environments making it impossible to replace their batteries. Consequently, the wireless sensor network is susceptible to failure due to the power depletion of some of its nodes. Therefore, a power constraint is identified as one of the most critical challenges in wireless sensor networks. Most of sensor node power is consumed in communication, as much as 80% of the total power [2]. Hence, techniques that aim toward reducing power consumption in wireless sensor networks are necessary to extend their expected life.

1.1 Characteristics and Challenges of Wireless Senor Networks

Wireless sensor networks requirements vary base on their main functionality and the phenomena being observed. This affects the number of nodes inside the network and its topology. So an understanding of the characteristics and challenges of wireless sensor networks is essential to build an optimal network. These characteristics and challenges are discussed in [1, 3, 4].

1.1.1 Cost

The production cost of a sensor node or a mote should be inexpensive, since the cost of the network is determined by the number of nodes used in this network. So to justify using wireless sensor network, a motes price is expected to be less than one U.S. dollar.

1.1.2 Power Consumption

Power consumption is an important metric for long-term operation of wireless sensor networks. Each node has a limited resource of energy, usually a battery. Other sources that extract energy from the surrounding environment can be used, i.e. solar cells. The network lifetime is determined by the lifetime of its nodes. These nodes should be as energy-efficient as possible. In an ad-hoc network, nodes work as routers where they forward packets from the source to the destination aside from their original functionality. Any failure in a group of nodes could cause a change in the topology and/or disconnectivity in some parts of the network. Studies showed that around 80% of a nodes energy is used in communication. Therefore, power management plans are needed to extend the expected life of wireless sensor networks

1.1.3 Communication Medium

For most environments where phenomena are being monitored by wireless sensor networks, it is difficult to implement a communication infrastructure. Therefore, nodes have to communicate by using wireless medium over multiple hops. This medium can be radio, infrared, or optical. The most commonly used medium is the radio for its low production value and ease of implementation. The wireless medium is known for its broadcast nature and superposition property. So, to transfer information from one node to another out of its reach over the network requires cooperation between the networks nodes. This ensures connectivity which is a global feature of interest.

1.1.4 Network Topology

Depending on the phenomena being observed, wireless sensor networks can be deployed randomly in the environment being observed or they can be placed individually according to a network design. In both cases topologies are subject to changes due to malfunction of sensor nodes, change in sensors position, or out of reachability. This requires a plan for topology maintenance.

1.1.5 Fault Tolerance

Sensor nodes are subject to failure due malfunction, noise, or obstacles. The failure should not affect the performance of the wireless sensor network or its functionality; this is known as network fault tolerance or reliability. In [5] the Poisson distribution is used to model the reliability $R_k(t)$ to define the probability of no failure during time interval $(0, t)$ in node k as:

$$R_k(t) = e^{\lambda_k t} \tag{1.1}$$

where t is the time period and λ_k is the failure rate.

1.1.6 Distributed Sensing and Processing

By distributing a large number of nodes in the environment being observed, more data can be collected about the environment. This leads to better coverage than using fewer sensing nodes with larger range where obstacles may exists. Since a large number of data can be collected, sensor nodes might have to process these data in addition to their neighbor nodes data in order to compute a function, i.e. max, min, and aggregate it to the sink node.

1.1.7 Scalability

Sensor nodes are deployed in large numbers in the environment being observed. Therefore, mechanics that help working with a large number of nodes have to be addressed. Corresponding to [6], a given area A with n nodes scattered in this area has a density of

$$\mu(R) = (n\pi R^2)/A \tag{1.2}$$

where R is the radio transmission radius of a node, $\mu(R)$ is the number of nodes inside this transmission radius.

1.2 Wireless Sensor Network Architecture

For wireless sensor networks to be able to monitor a given environment precisely, large number of senor nodes has to be deployed. As shown in Figure 1.2, each of these sensor nodes consists of four major components: a power unit, a communication unit, a sensing unit, and a processing unit. The power unit has to power the node and it consists of a battery and dc-to-dc converter. The sensing unit consists of one or multiple sensors and whose function is to collect data from the surrounding environment. The processing unit process the data

obtained from the sensors and other nodes. It consists of an analog-to-digital (AD) converter, a memory, and a microprocessor. The communication unit usually uses short range radio signals to communicate and aggregate data from one node to another and towards the sink node. The general architecture of a wireless sensor network is depicted in Figure 1.1. The

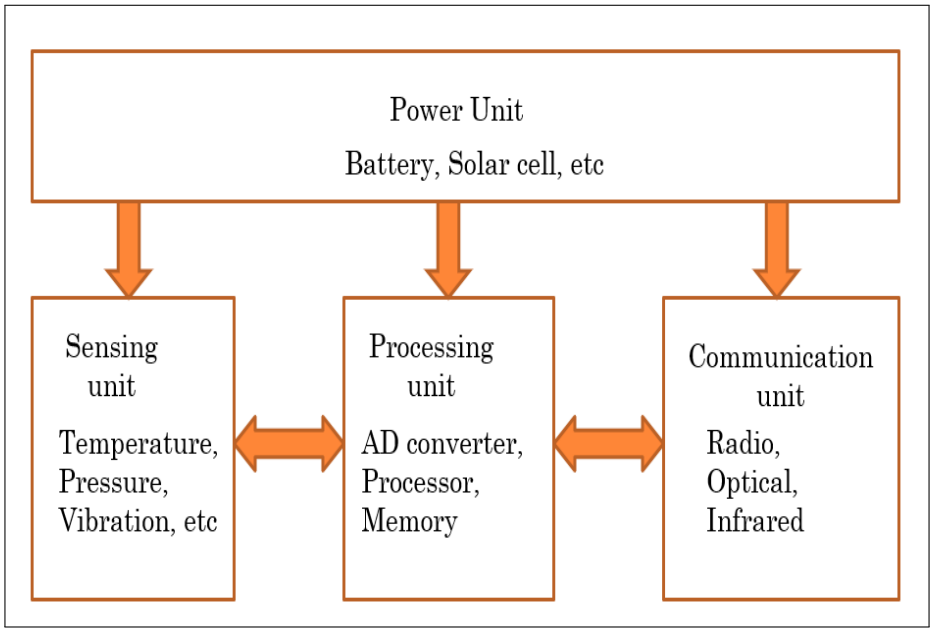

Figure 1.2: Structure of sensor node.

scattered nodes inside the environment start collecting data about the phenomena being observed. Then nodes start aggregating processed data toward the sink node. This method is known as data aggregation or data fusion and is used frequently in data centric networks. After the data arrives at the sink node, data will be forwarded to the central unit or the main server through the Internet. At the central unit analyses are applied on the data to extract certain features and parameters. These parameters can be used to update the network or for taking further actions if necessary.

1.3 Connectivity

In order for wireless sensor networks nodes to transfer the collected data back to the sink node they should maintain connectivity. Therefore connectivity is an important property of any network. Sensor nodes communicate with each other over multi hops through a wireless medium. Because of sensor nodes power constrains, sensor nodes usually form an ad-hoc network with arbitrary topology. This makes it hard for network nodes to maintain connectivity. Two major factors affecting the connectivity at each node are the transmission range of a node and the number of neighboring nodes (density of nodes in the environment).

For a node to expand its transmission range and its number of neighbors, it has to increase its transmission power. Both of these factors are locally chosen by each node based on the amount of power available for its operation. The wireless medium is known for its broadcast nature, attenuation, superposition, and node interference. Due to this nature it is difficult to design and/or analyze the process of transferring information over the wireless network. The problem of connectivity lies in establishing a network that is connected with high probability. The authors of [7,8] model wireless sensor networks as a random geometric graphs, then evaluate the critical range at which the network will be connected with high probability, assuming all nodes have the same transmission range. In [7], several definitions of connectivity are briefly summarized as follows:

- Definition 1: A network is connected if there is a path exists between any two nodes in the network. A network is m degree connected if there are m independent paths existing between any two nodes.
- Definition 2: If every node in the network is within the transmission range of another node then the network is connected.
- Definition 3: If a node x has m neighboring nodes, then the node degree is defined as $d(x) = m$. The node is said to be connected.

In [7], the random geometric graph is described as: a disk D has a unit area in r^2 where n nodes are placed uniformly and independently. This results in a graph $G(n, R(n))$ formed with a path existing between two nodes if they are within the transmission range, i.e. $R(n)$, of each other. Then the problem is to determine a transmission range that ensures the probability of a connected graph goes to one as n goes to infinity. This problem is examined in [8–10]. In [8], for a network to be connected the condition $p(n)R^2(n) \sim log(n)/n$ should be satisfied, where $p(n)$ is the probability of nodes keeping active status. In [9, 10], their result was summarized in [7] if $\pi R^2(n) = (log(n) + c(n))/n$ is chosen then the probability $G(n, R(n))$ is connected converges to one as n goes to infinity if and only if $c(n)$ goes to infinity. The previous discussion concerns the transmission range. The effect of the number of neighboring nodes on connectivity was studied in [11] and their results were summarized in [7]. Consider the network where each node is connected to its μn neighbors defined as $G(n, \mu n)$. Then there exist constants $c_1 > c_2 > 0$ such that $G(n, \mu n)$ is connected with a probability that goes to one if $\mu n \geq c_1 logn$ and disconnected with a probability that goes to one if $\mu n \leq c_2 logn$, as n goes to infinity. In a later study, these two constants can be less than one. From the above analyses the following results can be concluded. For a network to stay connected, each node should have a minimum transmission range sufficient enough to

maintain connectivity in between the networks nodes. It can also be concluded that a small number of neighboring nodes are needed to maintain this connectivity.

1.4 Capacity

Wireless sensor networks are distributed multi hop ad-hoc networks. Network nodes transfer data between each other cooperatively using a wireless channel. The nature of cooperation between nodes can be modeled in several ways. The simplest model is known as packet forwarding or packet relaying. All data transmitted over the channel endure attenuation and noise. In general this affects the probability of receiving the data correctly and results in consuming more power in order to retransmit the data again. So an important property of sensor networks is capacity. Capacity is concerned in finding how much information can be transmitted over the network reliably while maintaining low interference between network nodes. It is also concerned with the effect of increasing the number of nodes on the amount of information that can be transmitted over the network. Therefore, computing the exact capabilities of a wireless sensor network is almost impossible. In [7], boundaries on the capacity are presented by using two interference models, protocol and physical interference. The definitions of these models follow.

- Protocol model: Each node has a transmission range and interference range larger than its transmission range. Then each active transmission from x to y results in an interference range of radius $(1 + \Delta)d_{xy}$ centred around x, where $\Delta > 0$ and d_{xy} is the distance between x and y. A successful transmission if y is within the transmission range of x and outside the interference range of other active nodes. This is illustrated in Figure 1.3.

- Physical model: A successful transmission from x to y is defined if the signal-to-interference-plus-noise (SINR) is larger than a given threshold δ.

$$SINR(y) = \frac{P_x d_{xy}^{-\alpha}}{\sum_{k \in T \setminus \{x\}} P_k d_{ky}^{-\alpha} + \gamma} \geq \delta \tag{1.3}$$

  where P_x is the transmission power of x, α is the path loss exponent, and γ is the noise power.

By taking advantage of the large number of nodes in the network, modeling the network as a geometric random graph, and assuming a simple packet forwarding model, the capacity and its related issues, such as the optimal configuration of the network, are addressed using two metrics, transport capacity C_T and throughput capacity $C_T H$. Transport capacity measures

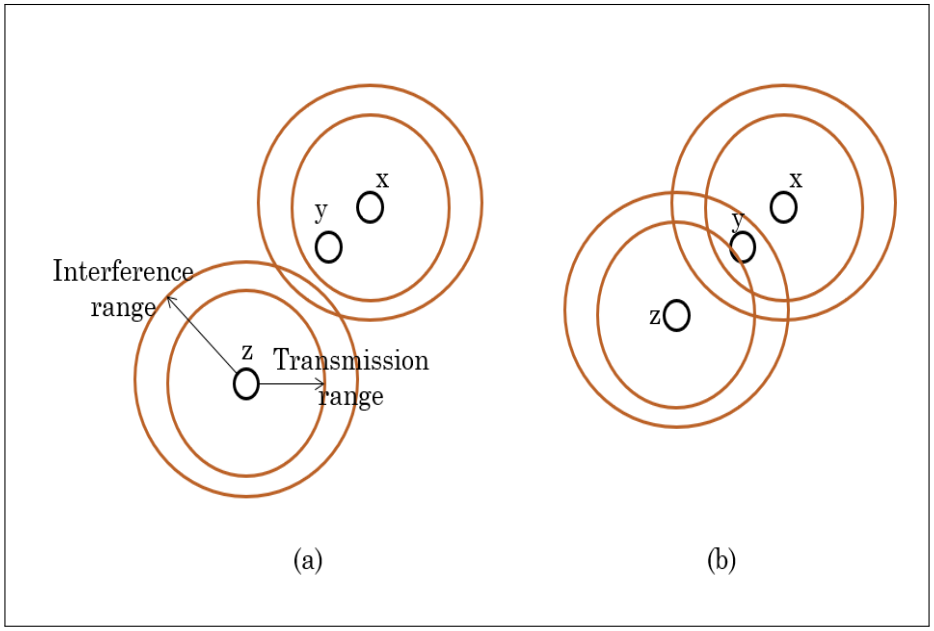

Figure 1.3: (a) Successful transmission. (b) Unsuccessful transmission.

how many meters a bit is transferred toward its destination during one second and it is measured in bit-meters/s. While throughput capacity is the maximum common throughput each node has inside the network. A major difference between these two metrics is that the former takes distance into consideration while the later does not The capacity limits in terms of transport capacity and throughput capacity is computed for both models in [12, 13] and summarized in [7]. In an optimal network when the protocol model is used $C_T = \Theta(\omega\sqrt{An})^2$ and for each node

$$\begin{aligned} \sqrt{\frac{An}{\pi}} \frac{\omega}{\sqrt{1+\Delta\sqrt{\Delta(\Delta+2)}}} &\leq C_T \\ &\leq \sqrt{\frac{8An}{\pi}} \frac{\omega}{\sqrt{1+\Delta\sqrt{\Delta(\Delta+2)}}} \end{aligned} \tag{1.4}$$

For the physical model, $C_T = \emptyset(\omega\sqrt{An})$ for all networks and $C_T = \Theta(\omega\sqrt{An})$ for optimal network. For the protocol model $C_{TH}(n) = c_2\omega/\sqrt{nlogn}$ is feasible while $C_{TH}(n) = c_1\omega/\sqrt{n\log n}$ is not. For the physical model $C_{TH}(n) = c_2\omega/\sqrt{n\log n}$ is feasible and $C_{TH}(n) = c_1\omega/\sqrt{n}$ is not, where $0 < c_2 < c_1 < \infty$, A is the area, and ω is node throughput bits/second. From these results [12] concluded an optimal scheme. This scheme suggests to group nodes into clusters with one head node for each cluster. Also [12], concludes that it is optimal for all nodes to use the same transmission range while maintaining connectivity.

1.5 Problem Statement and Contribution

The wireless medium is essentially unreliable and unpredictable, its links have higher high bit error rate, their characteristics vary over short time scales, and they are susceptible to interference due to their broadcast nature. Moreover, wireless medium is characterized as mobile which could result in dead spots and make it impossible to maintain full network connectivity. Therefore, introducing algorithms and mechanisms that reduce energy consumption is a vital key in prolonging the wireless sensor network life. Hence, maintaining a reliable and acceptable communication links while reducing energy consumption is foremost the most important aspect of wireless sensor networks' design.

In this thesis, we explore and aim to design a cheap wireless sensor network that consumes the least amount of energy and yet it provides a reliable communication medium. The contribution of this thesis is summarized as follow:

1. Modified Reed-Solomon Decoding algorithm:

   We proposed a novel approach to error correction codes in wireless sensor network. We introduce a modification to Reed-Solomon decoding algorithm which allows errors to occur in data without sacrificing the total integrity of the data. We showed that by deploying such mechanisms, we can reduce the total energy required to deliver data at their destination by reducing the decoding energy per symbol/bit. However, we concluded that the savings is minimal taken into account the degradation occurs in performance. As well, the requirement for having a non-binary error correction codes to achieve and replicate savings in decoders. As of now, the only well established non-binary error correction codes are Reed-Solomon codes.

2. Diversity Opportunistic Network Coding with Cooperation:

   Opportunistic network coding was introduced as mean to increase network throughput by XORing two or more packets together. However, it has been shown that ONC's bit error rate (BER) and throughput suffer when implemented in wireless network with noise model. We proposed a modification on opportunistic network coding (ONC) using diversity coding and cooperation, as well as, limiting the number of packets that can be network-coded together to three and only encode packets that were received by relay nodes directly. We showed that using such techniques we can alleviate the issues that plague ONC when implemented in noisy networks. We studied the effect of link outages/mobility on proposed solution and show that our proposed solution can accommodate up to one link failure.

3. Post-Quantum Hybrid Security Mechanism for MIMO Systems:

We studied the security of ad-hoc networks and propose a post-quantum hybrid security mechanism. We propose a security mechanism that take advantage of the wireless medium hereditary nature and cryptography techniques. This state of art protocol is able to overcome the presence of adversary eavesdropper and address man in the middle attack. Our security mechanism uses a combination of physical layer and cryptographic security techniques to provide best effort security.

1.6 Dissertation Outline

The rest of this dissertation is organized as follow: Chapter 2 describes the modified Reed-Solomon decoding algorithm. In Section 2.1, we discuss the basic elements of Reed-Solomon codes and provide an overview of the encoder and decoder. We propose our modification to RS decoder in Section 2.2. Section 2.3 describes the system model and simulation setup. In Section 2.4, we evaluate our proposed algorithm in term of bit error rate and power consumption. We conclude our discussion in Section 2.5. In chapter 3, we propose a diversity network coding with cooperation mechanism. Section 3.1 presents the background on network coding and discuss specifically ONC, as well as, diversity coding, and cooperative communication. In section 3.2, we propose our solution for network coding. We describe the system model and setup in section 3.3. We present the simulation results and provide discussion in section 3.4. Section 3.5, concludes our discussion in section 3.6. Chapter 4, discusses the post-quantum hybrid security mechanism for MIMO systems. In section 4.1, we provide background on cryptography and MIMO precoding. Section 4.2, describes the system model while section 4.3 details the proposed algorithm. Section 4.4 and 4.5, gives an account of security and performance analysis,respectively. We summarize our discussion in section 4.6. In chapter 5, we draw to close the dissertation by summarizing its contributions and propose future research directions to reduce energy consumption in ad-hoc networks generally and in wireless sensor networks specifically.

Chapter 2: Modified Reed-Solomon Decoding Algorithm

Communicating data reliably from one node to another toward the sink node is the foremost design aspect of wireless sensor networks. Thus, providing a reliable communication channel that consumes the least power amount possible becomes essential. A classical way of doing so is to use error control codes (ECC), such codes proved continuously their abilities in improving communication links reliability and decreasing the required transmission power. ECC append redundant bits or symbols to the original data to increase data resilience towards errors due to sensors' circuitry and the wireless medium. The appended bits or symbols later can be used to detect or correct errors at the decoder. ECC algorithms provide multiple advantages ranging from reducing bit error rate/symbol error rate (BER/SER), and reduce data retransmission. However, these advantages come at extra cost of processing power consumed in the encoder and the decoder. In several applications of wireless sensor networks, data can tolerate errors to some degree without risking their integrity. Imagine a weather forecast sensor network that reports temperature values to the main server periodically, if an error occurred and changed the actual value slightly, will this have a huge impact on the integrity of the system? No, most likely. Current ECC provide protection to the data over the network and detect/correct errors if they exist in any part of the data. The degree of how much error can be tolerated is determined based on application requirements. Consequently, modified ECC algorithms that allow such behavior should be developed and investigated.

We developed a modified version of RS decoding algorithm that allows errors to occur in data without correction [14]. To the best of our knowledge, this is the first time that data sensitivity and error location is taken into consideration in designing error correction techniques. All previous work investigated ECC algorithms that take into account correcting all bits/symbols despite of their location/importance [15], [16], [17], [18]. Other researchers used unequal ECC algorithms to provide some bits/symbols with extra protection yet they still correct errors despite their locations [19], [20]. In [21], multiple ECC algorithms were evaluated and shown to improve performance when compared to other techniques such as automatic repeat request (ARQ). The latter technique depends on acknowledging the reception of the packet and re-transmitting it in case of failure. It turns out that it consumes much more energy than that consumed in correcting the errors. We analyzed the performance and the power consumption of the modified RS decoding algorithm over a multi-hop additive white Gaussian noise (AWGN) channel and Rayleigh fading channels using multiple RS codes.

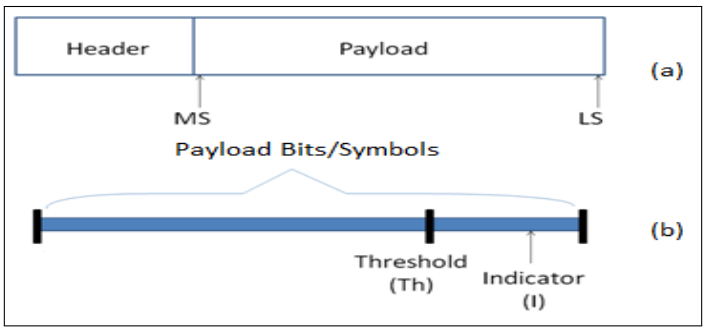

Figure 2.1: (a) Network data packet. (b) Payload bits/symbols with an indicator and threshold for the proposed ECC scheme.

Basically, the algorithm uses an indicator I to identify the error location(s) and a threshold Th to measure the error impact on the correctness of the received data. That is, if the indicator I exceeds a certain threshold Th then the packet needs to be corrected, otherwise, the packet is accepted and considered as a valid packet. The threshold Th placement is left to the application developer based on the application's data sensitivity, see Fig. 2.1 [14].

2.1 Background

RS codes were first introduced in 1960 by Irving S. Reed and Gustave Solomon at MIT. RS codes are block codes, where data message is divided into symbols. These codes are constructed over a finite field GF (2^m); where m is number of bits per symbol. They are a subclass of the non-binary cyclic error-correcting codes in which codewords can be generated by adding two or more codewords, or shifting its symbols. The RS (n, k) notation is used to represent RS codes that consist of n symbols, k information symbols, and n-k parity symbols. These codes can correct up to t errors, where t is defined by $\frac{(n-k)}{2}$ and known as the error correcting capability of ECC. Additionally, the minimum distance of these codes are defined as n-k+1. The structure of RS codewords is depicted in Fig. 2.2. The parity symbols are appended to the right or to the left of the information symbols; therefore, they are also known as systematic codes. RS codes are suitable for wireless communication channels with erasures and burst errors nature. Nowadays, they are widely used in digital communication systems and digital storage.

2.1.1 Galios Field

Galois fields (GF) have many applications in coding theory. The symbols in a Reed-Solomon code are elements of GF. Since GF consists of a finite set of elements, RS codes can be represented by a fixed length codewords. A GF defined as GF(p^m), where p is a prime number

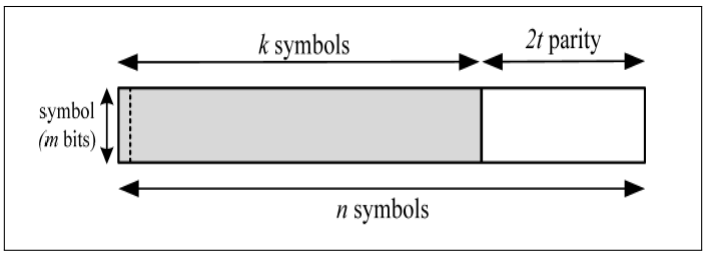

Figure 2.2: RS codeword structure.

and m is an integer, are compromised of polynomials of degree $m-1$. These polynomials are expressed as in equation (2.1) and their coefficients take on values in the set 0,1,..$p-1$.

$$a_{m-1}x^{m-1} + ... + a_1x + a_0 \tag{2.1}$$

The elements in GF are based on a primitive element, denoted α and the elements of a GF can be represented in index form as: (2.2).

$$0, \alpha^0, \alpha^1, ..., \alpha^{N-1} \tag{2.2}$$

When deployed in coding p is commonly set to be 2. This construct a field denoted as GF(2^m) which consists of 2^m elements. If the primitive element equals 2, the power N in (2.2) will then be $N = 2^m$-1. The coefficients $a_0, a_1, ..., a_{m-1}$ in (2.1) take on values of 0,1 thus its elements can be represented as a binary number. 2^m elements in GF, can then be represented as combinations of a m-bit number. If $m = 4$ the Galois field is GF(2^4) or GF(16). This field has 16 elements, and can be represented by a 4-bit number from 0000_b to 1111_b or 0_d to 15_d.

2.1.2 Field Generator Polynomial

A Galois field can be constructed using a field generator polynomial or primitive polynomial. This polynomial is the minimal polynomial of a primitive element of the finite extension field GF(p^m). The primitive polynomial $p(x)$ is of degree m and is irreducible, meaning it has no factors [22]. The primitive element is a root of $p(x)$. By using this, all non-zero elements of GF(p^m) can be constructed using a successive power of α. A field might have several primitive polynomials, and each primitive polynomial give a unique representation of its elements. For instance, the field GF(16) have two primitive polynomials, $p(x) = x^4+x+1$ and $p(x) = x^4 + x^3 + 1$ [23]. To illustrate the construction of GF(16), the prior primitive polynomial will be used as an example.

When constructing the field, the primitive polynomial is set equal to zero, $p(\alpha) = 0$. This can be done because the primitive element is a root of the primitive polynomial. Then the primitive polynomial can be written as $p(\alpha) = \alpha^4 + \alpha + 1 = 0$ which can be rewritten as $\alpha^4 = a + 1$.

To construct the whole field in polynomial form, α is multiplied at each stage. When the polynomial form reaches $\alpha^4, \alpha + 1$ is substituted in its place. The resulting terms is finally added together using Galois field addition. The first five elements of GF(16) in polynomial form are $0, 1, \alpha, \alpha^2, \alpha^3$, and rest of the non-zero elements in GF(16) are found in the following way:

$$\alpha^4 = \alpha + 1$$
$$\alpha^5 = \alpha(\alpha^4) = \alpha(\alpha + 1) = \alpha^2 + \alpha$$
$$\alpha^6 = \alpha(\alpha^5) = \alpha(\alpha^2 + \alpha) = \alpha^3 + \alpha^2$$
$$\alpha^7 = \alpha(\alpha^6) = \alpha(\alpha^3 + \alpha^2) = \alpha^4 + \alpha^3 = \alpha^3 + \alpha + 1$$
$$\alpha^8 = \alpha(\alpha^7) = \alpha(\alpha^3 + \alpha + 1) = \alpha^4 + \alpha^2 + \alpha = \alpha^3 + \alpha + \alpha + 1 = \alpha^2 + 1$$
$$\alpha^9 = \alpha(\alpha^8) = \alpha(\alpha^2 + 1) = \alpha^3 + \alpha$$
$$\alpha^{10} = \alpha(\alpha^9) = \alpha(\alpha^3 + \alpha) = \alpha^4 + \alpha^2 = \alpha^2 + \alpha + 1$$
$$\alpha^{11} = \alpha(\alpha^{10}) = \alpha(\alpha^2 + \alpha + 1) = \alpha^3 + \alpha^2 + \alpha$$
$$\alpha^{12} = \alpha(\alpha^{11}) = \alpha(\alpha^3 + \alpha^2 + \alpha) = \alpha^4 + \alpha^3 + \alpha^2 = \alpha^3 + \alpha^2 + \alpha + 1$$
$$\alpha^{13} = \alpha(\alpha^{12}) = \alpha(\alpha^3 + \alpha^2 + \alpha + 1) = \alpha^4 + \alpha^3 + \alpha^2 + \alpha = \alpha^3 + \alpha^2 + \alpha + \alpha + 1 = \alpha^3 + \alpha^2 + 1$$
$$\alpha^{14} = \alpha(\alpha^{13}) = \alpha(\alpha^3 + \alpha^2 + 1) = \alpha^4 + \alpha^3 + \alpha = \alpha^3 + \alpha + \alpha + 1 = \alpha^3 + 1$$

2.1.3 Addition and Subtraction in Galois Field

Addition and subtraction in Galois field are done in exactly the same way using a exclusive-OR function (XOR), or by modulo 2 addition/subtraction of the coefficients [23]. Since addition and subtraction have exactly the same effect, addition is used when performing a

Table 2.1: Field elements for GF(16) with $p(x) = x^4 + x + 1$

Index Form	Polynomial Form	Binary form	decimal form
0	0	0000	0
α^0	1	0001	1
α^1	α	0010	2
α^2	α^2	0100	4
α^3	α^3	1000	8
α^4	$\alpha + 1$	0011	3
α^5	$\alpha^2 + \alpha$	0110	6
α^6	$\alpha^3 + \alpha^2$	1100	12
α^7	$\alpha^3 + \alpha + 1$	1011	11
α^8	$\alpha^2 + 1$	0101	5
α^9	$\alpha^3 + \alpha$	1010	10
α^{10}	$\alpha^2 + \alpha + 1$	0111	7
α^{11}	$\alpha^3 + \alpha^2 + \alpha$	1110	14
α^{12}	$\alpha^3 + \alpha^2 + \alpha$+1	1111	15
α^{13}	$\alpha^3 + \alpha^2 + 1$	1101	13
α^{14}	$\alpha^3 + 1$	1001	9

subtraction operation. In polynomial form this is written as shown in (2.3).

$$\sum_{i=0}^{m-1} a_i x^i + \sum_{i=0}^{m-1} b_i x^i = \sum_{i=0}^{m-1} c_i x^i \tag{2.3}$$

Since addition is a XOR operation and the coefficients can only take the value 0 or 1, $c_i = 0$ when $a_i = b_i$ and $c_i = 1$ when $a_i \neq b_i$ for $0 \leq i \leq m-1$.If we want to add the numbers 10 and 14 in GF(16) this will result in 4. Using a polynomial expression this results in $(x^3 + x) + (x^3 + x^2 + x) = x^2$

2.1.4 Multiplication and Division in Galois Field

When multiplying two polynomials with degree $m - 1$, the resulting product polynomial would have a degree of $2m - 2$. In Galois field multiplication the product can not be larger than the largest element of the field GF(2^m), thus multiplication in Galois field is defined

as the product modulo the field generator polynomial $p(x)$ [23]. The product modulo can be found by dividing the product polynomial by the field generator polynomial $p(x)$, and then take the remainder. This will always give a result that is inside the Galois field. There are several different ways in which the remainder can be found. One possible way is to first multiply the values using the polynomial expression, and then divide the result by the field generator polynomial. This division is done by multiplying the divisor by a value to make it the same degree as the dividend, and then subtracting the divisor from the dividend [23]. For example, to multiply the two values 12 and 15 in Galois field GF(16). First, we multiply the two values using the polynomial expression. Second, we use Galois addition on the values with the same exponents, as shown below.

$$\begin{aligned}(x^3 + x^2)(x^3 + x^2 + x + 1) &= x^6 + x^5 + x^4 + x^3 + x^5 + x^4 + x^3 + x^2 \\ &= x^6 + x^2\end{aligned}$$

Then the result is divided by the field generator polynomial.

$$\begin{array}{r|l} & x^2 \\ \hline x^4 + x + 1 & x^6 \quad + x^2 \\ & -x^6 - x^3 - x^2 \\ \hline & -x^3 \end{array}$$

The resulting remainder is then the product of the two values. In the example above the remainder turn out to be x^3 which is equal to 1000_b or 8_d.

Dividing two elements in a Galois field can be achieved by multiplying by the inverse of the divisor. The inverse of a field element is defined as the element value, that when multiplied by the field element produces a value of 1. Below is an example on how this can be done. We want to divide 15 by 12. First the inverse of 12 is found, which is:

$$\begin{aligned}12 &= \alpha^6 \\ \alpha^{(-6) mod 15} &= \alpha^9 = 10 \\ 15/12 &= 15 \times 10\end{aligned}$$

Then 15 is multiplied by 10 to get the result. This can be done using the multiplication technique previously described in this section.

$$\begin{aligned}(x^3+x^2+x+1)(x^3+x) &= x^6+x^4+x^5+x^3+x^4+x^2+x^3+x \\ &= x^6+x^5+x^2+x\end{aligned}$$

$$\begin{array}{r|l} & \quad\quad\quad\quad\quad\quad\quad\quad x^2+x \\ \cline{2-2} x^4+x+1 & \quad x^6+x^5 \quad\quad\quad +x^2+x \\ & \quad -x^6 \quad\quad -x^3-x^2 \\ \cline{2-2} & \quad\quad\quad x^5-x^3 \quad\quad +x \\ & \quad\quad -x^5 \quad\quad -x^2-x \\ \cline{2-2} & \quad\quad\quad\quad -x^3-x^2 \end{array}$$

Dividing 15 by 12 results in 1100_b or 12_d.

2.1.5 RS Encoder

The generator polynomial construction for Reed-Solomon codes is the approach most commonly used today in the error control literature. This approach initially evolved independently from RS codes as a means for describing cyclic codes. A code is said to be cyclic if, for any code word $\boldsymbol{C} = (C_0, C_1, C_2, ..., C_{n-2}, C_{n-1})$, the cyclically shifted word $\boldsymbol{C'} = (C_1, C_2, ..., C_{n-2}, C_{n-1}, C_0)$ is also a codeword. Gorenstein and Zierler then generalized Bose and Ray-Chaudhuri's work to arbitrary GF(p^m), discovering along the way that they had developed a new means for describing Reed and Solomon's "polynomial codes" [8]. If an (n, k) code is cyclic, it can be shown that the code can always be defined using a generator polynomial $g(x) = g_0 + g_1x + g_2x^2 + .. + g_{n-k}x^{n-k}$. In this definition each codeword is interpreted as a code polynomial.

$$(C_0, C_1, C_2, ..., C_{n-2}, C_{n-1}) \rightarrow C_0 + C_1x + C_2x^2 + .. + C_{n-1}x^{n-1}$$

A vector $\boldsymbol{C}$ is a code word in the code defined by $g(x)$ if and only if its corresponding code polynomial $C(x)$ is a multiple of $g(x)$. This provides a very convenient means for mapping information symbols onto code words. Let $\boldsymbol{M} = (M_0, M_1, ..., M_{k-1})$ be a block of k information symbols. These symbols can be associated with an information polynomial

$M(x) = M_0 + M_1x + ... + M_{k-1}x_{k-1}$, which is encoded through multiplication by $g(x)$.

$$C(x) = M(x)g(x)$$

Cyclic RS codes with code word symbols from GF(q) have length $q-1$. As the generator polynomial approach to constructing. Reed-Solomon codes is currently the most popular, RS codes with symbols in the field GF(q) usually have length $q-1$. The Reed-Solomon design criterion is as follows: The generator polynomial for a t-error-correcting code must have as roots $2t$ consecutive powers of α.

$$g(x) = \prod_{j=1}^{2t}(x + \alpha^j)$$

To produce a codeword in systematic form where parity symbols is appended to the message. The encoder shifts the message polynomial by multiplying it with x^{n-k} and the result is divided by $g(x)$. This results in a quotient $q(x)$ and a remainder $r(x)$.

$$\frac{M(x)x^{n-k}}{g(x)} = q(x) + \frac{r(x)}{g(x)}$$

Then the codeword consists of $M(x)$ and $r(x)$ as follow.

$$C(x) = M(x)x^{n-k} + r(x)$$

2.1.6 RS Decoder

A transmitted codeword $C(x)$ is susceptible for errors and can get corrupted while being transmitted to its destination due to multiple factors. These errors can be represented as an error polynomial $E(x)$ and affect the symbols in their perspective position. Therefore, a transmitted codeword received by the decoder can be written as follow:

$$R(x) = C(x) + E(x) \tag{2.4}$$

A typical RS decoder consists of five stages, each of these stages can be represented algebraically and can be processed either serially or parallelly. First, the decoder start by calculating the syndromes which then allows the decoder to determine the errors locations and magnitudes. Using the key equation the error locator polynomial and error evaluator polynomial are found. In this work the Berlekamp-Massey algorithm is utilized to determine the error locator polynomial and hence the errors' location. After that, the two polynomials

are used to find the errors' values using Forney algorithm. Finally, the decoder corrects the errors in the received codeword. Figure 2.3 shows the RS decoder structure, each of these units is detailed next.

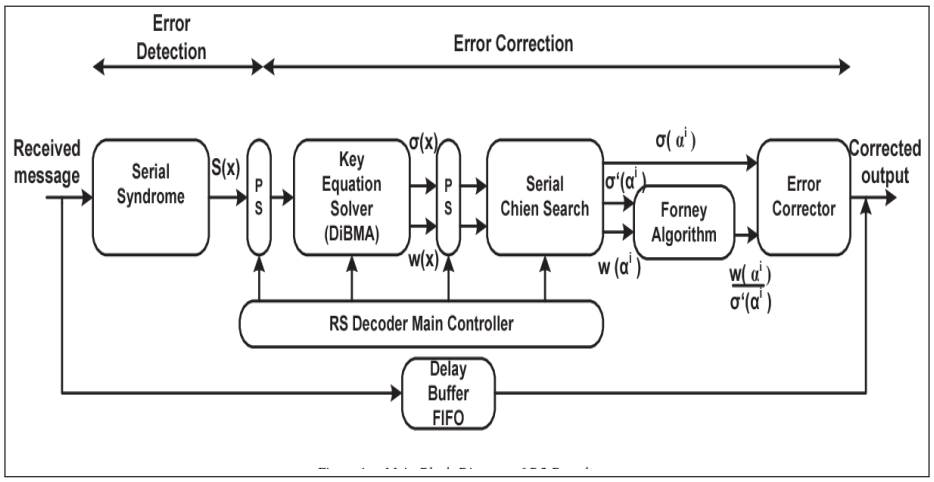

Figure 2.3: RS decoder structure.

2.1.7 Syndrome Calculation

The first step for the RS decoder to correct an erroneous codeword is to calculate the syndrome. The syndrome consists of $2t$ values and is only dependent on the error polynomial, hence, a codeword with no errors will have a syndrome values of zeros for all its $2t$ values. One way to calculate the syndrome is to divide the received codeword polynomial by the generator polynomial $g(x)$ which is equivalent to dividing the received codeword polynomial by its factors. This process can be described mathematically as shown in equation 2.5, where each syndrome value is denoted by S_i and index i is defined over $0 \leq i \leq 2t-1$.

$$\frac{R(x)}{g_i(x)} = Q_i(x) + \frac{S_i}{g_i(x)}, where \quad g_i(x) = (x + \alpha^i) \tag{2.5}$$

Alternatively, the syndrome can be calculated by substituting the roots α^i into the received codeword polynomial in it. Since adding the same values in GF results in zero, this

will results in $Q_i(x+\alpha_i)$=0. Then equation 2.5 can be rewritten as equation 2.6.

$$\begin{aligned} S_i &= Q_i(x)(x+\alpha^i) + R(x) \\ &= Q_i(\alpha^i)(\alpha^i+\alpha^i) + R(\alpha^i) \\ &= R(\alpha^i) \\ S_i &= R_{n-1}(\alpha^i)^{n-1} + R_{n-2}(\alpha^i)^{n-2} + ... + R_1(\alpha^i) + R_0 \end{aligned} \tag{2.6}$$

The syndromes can also be expressed as an syndrome polynomial as shown in equation 2.7.

$$S(x) = \sum_{i=0}^{2t-1} S_i x^i \tag{2.7}$$

Since $g(x)$ is factor of $R(x)$, S_i can be rewritten as:

$$S_i(\alpha^i) = R(\alpha^i) = E(\alpha^i) \tag{2.8}$$

This concludes that syndrome values are only dependent on errors introduced in the codeword and when there is no error the syndrome values equal to zero. Assuming v errors, where $v \leq t$:

$$\begin{aligned} E(x) &= Y_1 x^{e_1} + Y_2 x^{e_2} + ...Y_v x^{e_v} \\ S_i &= E(\alpha^i) \\ S_i &= Y_1\alpha^{ie_1} + Y_2\alpha^{ie_2} + ...Y_v\alpha^{ie_v} \\ &= Y_1 X_1^i + Y_2 X_2^i + ...Y_v X_v^i \end{aligned} \tag{2.9}$$

The $2t$ syndrome equations can be represented as:

$$\begin{bmatrix} S_0 \\ S_1 \\ \vdots \\ \vdots \\ S_{2t-1} \end{bmatrix} = \begin{bmatrix} X_1^0 & X_2^0 & \cdots & X_v^0 \\ X_1^1 & X_2^1 & \cdots & X_v^1 \\ \vdots & \vdots & \vdots & \\ \vdots & \vdots & \vdots & \\ X_1^{2t-1} & X_2^{2t-1} & \cdots & X_v^{2t-1} \end{bmatrix} \times \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ \vdots \\ Y_v \end{bmatrix} \tag{2.10}$$

2.1.8 Error Locator Polynomial

After calculating the syndrome and determining that errors occur in the received codeword the next step is to calculate the error locator polynomial. The equation below also known as key equation is often used to find the error locator polynomial and its evaluator. These two

Table 2.2:
Berlekamp-Massey algorithm parameters

μ	-1
σ^{μ}_{BM}	1
$d\mu$	1
$l\mu$	0
$\mu - l\mu$	-1

polynomials are used to determine error locations and values.

$$\Omega(x) = S(x)\sigma(x) mod x^{2t} \tag{2.11}$$

The error locator polynomial is described as:

$$\begin{aligned} \sigma(x) &= (1 + X_1 x)(1 + X_2 x ... 1 + X_v x) \\ &= 1 + \sigma_1 x + ... + \sigma_{v-1} x + \sigma_v \end{aligned} \tag{2.12}$$

To find the coefficients of the error locator polynomial the Berlekamp-Massey algorithm is utilized. The algorithm is described in Algorithm 1. The algorithm parameters are

Algorithm 1 Berlekamp-Massey algorithm

Input d_μ
if $d_\mu = 0$ **then**
  $\sigma^{\mu+1}_{BM}(x) = \sigma^{\mu}_{BM}(x)$
  $l_{\mu+1} = l_\mu$
else if $d_\mu \neq 0$ **then**
  $\sigma^{\mu+1}_{BM}(x) = \sigma^{\mu}_{BM}(x) + d_\mu d \rho^{-1} x^{(\mu-\rho)} \sigma^{\rho}(x)$
  $l_{\mu+1} = max(l_\mu, l_\rho + \mu - \rho)$
  $d_{\mu+1} = S_{\mu+2} + \sigma^{\mu+1}_1 S_\mu + \cdots + \sigma^{\mu+1}_{l_{\mu+1}} S_{\mu+2-l_\mu}$
end if

summarized in Table 2.2. After finding the error locator polynomial, it can be substituted into equation 2.11.

2.1.9 Chien Search

Chien search is used to determine the roots of the error locator polynomial by evaluating $\sigma(x)$ with all elements of the GF(2^m) α^i, where $0 \leq i \leq n-1$. If $\sigma(\alpha^i) = 0$ then it is a root and the error location is the inverse position of i.

2.1.10 Forney Algorithm

The Forney algorithm is used to find the error values. The Forney alforithm uses the error locator and the error evaluator polynomials to determine the value following the following equation, where $l = 1, 2, \cdots, v$.

$$Y_l = \frac{\Omega(X_l^{-1})}{\sigma\prime(X_l^{-1})} \tag{2.13}$$

Where $\sigma\prime(X_l^{-1})$ is the derivative of $\sigma(x)$ for $x = X_l^{-1}$.

2.2 Modified RS Decoding Algorithm

Since RS codes are non-binary, we choose to experiment with them. Contrary to binary codes, RS codes require locating errors' positions and then evaluating errors' magnitude to correct for errors. The requirement for evaluating errors' magnitude allows a room to reduce the amount of computation required and hence save energy. To decode RS codes, we used Berlekamp-Massey algorithm to determine errors' locations and Forney algorithm to determine errors' magnitude [24]. Although there are multiple algorithms to identify the errors' positions and magnitude, Berlekamp-Massey algorithm and Forney algorithm have been established to be the most efficient.

Both the original RS decoding algorithm and the modified RS algorithm pseudo codes are depicted in Fig. 2.4, and Fig. 2.5, respectively [14]. Based on what discussed previously, the modified algorithm uses the error locations as its indicators to determine if source symbols need to be corrected or not. If all errors occur to the right of Th, and number of errors (e) is less than or equal to t then data will be accepted (see Fig. 2.1). If all errors occur to the right of Th and e is greater than t then data are corrupted and cannot be corrected, therefore packet will be dropped. If any error occurs to the left of Th and e is less or equal to t then data will be corrected. If any error occurs to the left of Th and e is greater than t then data are corrupted and cannot be corrected, therefore packet will be dropped. Finally, the placement of Th is left for the network administrator or/and the application developer.

```
Algorithm: ORIGINAL_RS_DECODER (Received_DATA, n, k)

        Step 1: Evaluate the syndrome, 'S'
        if S==0
          comment
          No error occured within Received_DATA
          Received_DATA = [I, P]
          Truncate information symbols
          End_comment
          return (I)
        end_if

        Step 2: Find error location polynomial 'Sigma'
        Sigma ←Berlekamp_Massey_Algorithm (S, t)

        Step 3: Find the roots of Sigma
        Sigma_Roots ←Chin_Search (Sigma)
        if is (Sigma_Roots) not distinct?
          comment
          e>t
          Data are corrupted
          Drop information/ Request retransmission.
          end_comment
          return()
        end_if
        calculate: Reciprocals_Sigma_Roots
        comment: Error locations are determined

        Step 4: Calculate Error_Values and Corrected_DATA
        Error_Values ← Forney_Algorithm (Sigma, Reciprocals_Sigma_Roots)
        Corrected_DATA ← Received_DATA+ Error_Values
        comment: Corrected_DATA = [I', P']
        return (I')
end
```

Figure 2.4: Original RS decoding algorithm pseudo code.

```
Algorithm: MODIFIED_RS_DECODER (Received_DATA, n, k)

        Step 1: Evaluate the syndrome, 'S'
        if S==0
          comment
          No error occured within Received_DATA
          Received_DATA = [I, P]
          Truncate information symbols
          End_comment
          return (I)
        end_if

        Step 2: Find error location polynomial 'Sigma'
        Sigma ←Berlekamp_Massey_Algorithm (S, t)

        Step 3: Find the roots of Sigma
        Sigma_Roots ←Chin_Search (Sigma)
        if is (Sigma_Roots) not distinct?
          comment
          e>t
          Data are corrupted
          Drop information/ Request retransmission.
          end_comment
          return()
        end_if
        calculate: Reciprocals_Sigma_Roots
        comment: Error_Locations are determined

        Step 4: Compare error locations against threshold 'Th'
        if Error_Locations>= Th
          return (I);
        end_if

        Step 5: Calculate Error_Values and Corrected_DATA
        Error_Values ← Forney_Algorithm (Sigma, Reciprocals_Sigma_Roots)
        Corrected_DATA ← Received_DATA+ Error_Values
        comment: Corrected_DATA = [I', P']
        return (I')
end
```

Figure 2.5: Modified RS decoding algorithm pseudo code.

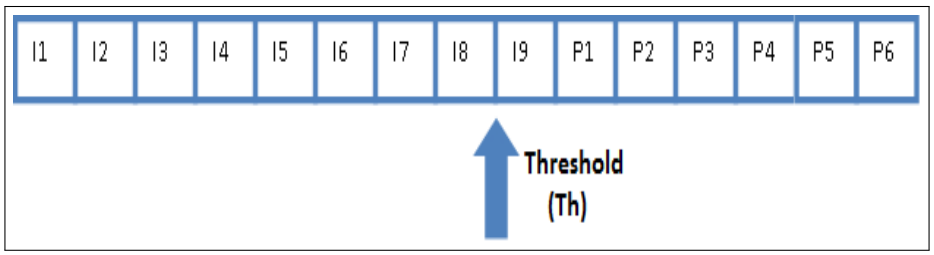

Figure 2.6: MRS(15,9) showing threshold placement.

2.2.1 Decoding Example

Assume the non-binary codeword from RS(15,9) over GF(2^4) was sent as follow:

$$\mathbf{c} = \mathbf{0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0}$$

Also, assume the received codeword as follow:

$$\mathbf{r} = \mathbf{0\,0\,0}\,1011\,\mathbf{0\,0}\,1000\,\mathbf{0\,0\,0\,0\,0}\,0011\,\mathbf{0\,0}$$

where $\mathbf{0} = 0000$. Then the received codeword can be rewritten as $\mathbf{r}(x) = \alpha^7 x^3 + \alpha^3 x^6 + \alpha^4 x^{12}$. The syndrome can be calculated as follow:

$$\begin{aligned}
S_1 &= \alpha^{10} + \alpha^9 + \alpha = \alpha^{12} \\
S_2 &= \alpha^{13} + 1 + \alpha^{13} = 1 \\
S_3 &= \alpha + \alpha^6 + \alpha^{10} = \alpha^{14} \\
S_4 &= \alpha^4 + \alpha^{12} + \alpha^7 = \alpha^{10} \\
S_5 &= \alpha^7 + \alpha^3 + \alpha^4 = 0 \\
S_6 &= \alpha^{10} + \alpha^9 + \alpha^{10} = \alpha^{12}
\end{aligned}$$

Using the Berlekamp-Massey algorithm the error location polynomial $\sigma(x)$ can be determined as shown in Table 2.3. In this example, $\sigma(x) = 1 + \alpha^7 x + \alpha^4 x^2 + \alpha^6 x^3$ and by substituting $1, \alpha, \ldots, \alpha^{n-1}$ into $\sigma(x)$ we find its roots. Consequently, the reciprocals of these roots are the error locations, namely x^3, x^6, and x^{12}. Since errors happens on both sides of the defined threshold the modified algorithm will attempt to correct all errors. Alternatively, if the error was found at x^{12} the modified algorithm will perceive the packet as correct and will not attempt to calculate the error magnitude nor correct the error. Figure 2.6 depicts the threshold placement in the example codeword.

Table 2.3: Finding error location polynomial using Berlekamp-Massey algorithm

μ	σ^{μ}	d_{μ}	l_{μ}	$\mu - l_{\mu}$
-1	1	1	0	0
0	1	α^{12}	0	0
1	$1 + \alpha^{12}x$	α^{7}	1	0
2	$1 + \alpha^{3}x$	1	1	1
3	$1 + \alpha^{3}x + \alpha^{3}x^{2}$	α^{7}	2	1
4	$1 + \alpha^{4}x + \alpha^{12}x^{2}$	α^{10}	2	2
5	$1 + \alpha^{7}x + \alpha^{4}x^{2} + \alpha^{6}x^{3}$	0	3	2
6	$1 + \alpha^{7}x + \alpha^{4}x^{2} + \alpha^{6}x^{3}$	-	-	-

2.3 System Model and Simulation Setup

A linear wireless sensor network topology with equal distance placement, d, of its nodes is considered here. This model is appropriate to study the effect of multi-hop packet forwarding which represents a common operation mode of data forwarding from an end-node to a sink node. In this paper, the modified RS algorithm along with the original RS algorithm and the un-coded data are evaluated over multi-hop AWGN channel and Rayleigh fading channel using BPSK modulation. For simulation purposes, we used a slow frequency-flat Rayleigh fading channel, i.e. a channel that consists of one non-line-of-sight path, with maximum Doppler frequency set to 1 Hz. Also, we set the number of hops to 6 to best reflect a realistic case of packet forwarding process.

The system model uses settings and methods like the ones found in [15] to determine the performance and the power consumption of each method. The system uses 2.4 GHz transmission frequency. The calculated bit error rate (BER) and symbol error rate (SER) are used to evaluate the performance while the following equations, found in [22], are used to calculate the power consumption.

$$SNR = P_{tx} - Attenuation - N_{Th} - NF_{rx} + G_{ECC} \tag{2.14}$$

$$Attenuation = 20\log_{10}(\frac{4\pi}{\lambda}) + 10n\log_{10} d \tag{2.15}$$

$$G_{ECC} = SNR_{UC} - SNR_{ECC} \tag{2.16}$$

where SNR is the signal-to-noise ratio in dB, P_{tx} is the transmission power required at a certain SNR to achieve a desired BER value in dB, N_{Th} is the thermal noise in dB, NF_{rx} is

the receiver noise figure in dB, G_{ECC} is the error control code gain resulting from the use of ECCs, λ is the wavelength, d is the distance between transmitter and receiver in meters, and n is the path loss exponent.

The G_{ECC} is determined by taking the difference between the SNR value of the un-coded data and the SNR value of a specific error control scheme that achieves a certain BER value, as shown in 2.16. To calculate P_{tx}, we first identify the desired BER value and then substitute values into 2.14.

The modified and the original RS decoding algorithms are evaluated using multiple RS codes. Specifically, RS(7,3), RS(15,9), and RS(15,5) are used in simulations. These codes can correct two, three, and five errors, respectively. The original RS decoding algorithm corrects errors despite of the locations of these errors. Alternatively, the modified RS decoding algorithm corrects errors to the left of Th. The Th is set in a way that allows errors to occur in parity symbols and the least significant information symbol.

2.4 Performance Analysis

We evaluate the performance and power consumption of the modified RS decoding algorithm, the original RS decoding algorithm, the un-coded data over multi-hop AWGN and Rayleigh fading channels using multiple RS codes. RS(7,3), RS(15,9), and RS(15,5) and the modified version of these codes MRS(7,3), MRS(15,9), and MRS(15,5) are used for evaluation. The performance is evaluated by computing the BER and SER for each method, BER/SER is defined as number of bits/symbols received in error divided by the total number of transmitted bits/symbols. The power consumption is estimated by calculating the transmission energy per bit required to achieve a certain BER, and the average decoding energy per bit. The energy saving gain for RS codes are determined by subtracting the energy required to deliver a bit at destination at certain BER from the energy required to deliver a bit at destination by un-coded data while achieving the same BER. Then, we estimate the energy saving per bit between RS codes and the modified version base on simulated power consumption of RS decoder's components.

2.4.1 BER and SER Performance Analysis

The multi-hop AWGN channel BER and SER for all three methods are depicted in Figs. 2.7 to 2.10. Fig. 2.7 and Fig. 2.9 reflect the actual BER and SER, respectively. On the other hand, Fig. 2.8 and Fig. 2.10 reflect the conditional BER and SER, respectively. We condition the BER/SER on the correctness of the significant bits/symbols, i.e. we only count the errors occurred to the left of Th. The notion of conditional BER/SER introduced here is to reflect

the BER/SER performance of the modified RS decoding algorithms since we argue here that error(s) occur to the right of Th shall not impact the integrity of the actual data by design. The notation RS and MRS used in the figures represents the performance of the original and the modified decoding algorithm, correspondingly. The results closely match the ones obtained in the single-hop scenario presented in [14]. This is expected since each node in the path decodes and corrects for errors making the process of packet forwarding from node to the next an iid process. Examining Fig. 2.7 and Fig. 2.9, we can see that the original RS decoding algorithm achieves the highest performance in comparison to the modified RS decoding algorithm and the un-coded data. This happens because the original RS decoding algorithm corrects all errors occur in a packet while the modified RS decoding algorithm only corrects errors to the left of Th as detailed in Section 2.2. However, taking into consideration that we allow errors to the right of Th and we do not demand the correctness of data to the right of Th. The modified RS decoding algorithm matches the original algorithm performance when BER and SER are conditioned on the correctness of data to the left of Th as shown in Fig. 2.8 and Fig. 2.10. It is worth noticing that MRS(15,9) achieves higher performance than MRS(15,5) at higher SNR values even though MRS(15,5) is capable of correcting more errors than MRS(15,9). This is because MRS(15,5) uses more parity symbols than MRS(15,9) and the modified decoding algorithm might leave parity errors without correction.

As in the AWGN case, the actual and the conditional BER and SER in the multi-hop Rayleigh fading channel are portrayed in Figs. 2.11 to 2.14. Examining Fig. 2.11 and Fig. 2.13, we can see that the original and the modified RS decoding algorithms have similar performance with the original RS decoding algorithm coming on top of the modified decoding algorithm. This is because packets are exposed to severe fading and most of the bits/symbols are in error. It is worthy to point out that both the original and the modified RS decoding algorithms achieve better SER than BER, as opposed to AWGN case. This is happen because RS codes are useful in wireless medium where burst errors are dominant in Rayleigh fading channel. Similar to the AWGN case, the conditional BER and SER depicted in Fig. 2.11 and Fig. 2.14 shows that both decoding algorithms achieve the same performance when taking into consideration the correctness of significant bits/symbols of data. It is expected that the margin between the original and modified decoding algorithms to be more clear if techniques such as interleaving, however, this is beyond the scope of this work. It is clear that the modified decoding algorithm has its merits and become more valuable when implemented with fading in mind. Finally, it is worth mentioning that the Rayleigh pdf parameter σ^2 is set to 0.5.

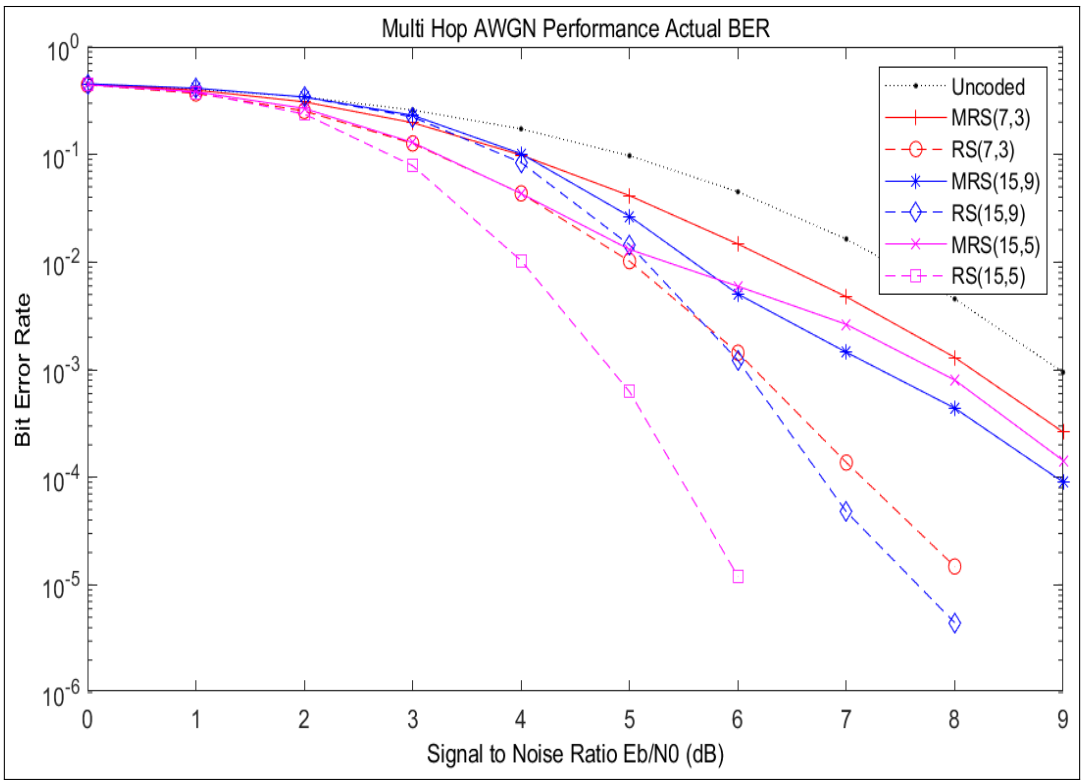

Figure 2.7: Actual bit error rate for un-coded data, modified RS code, and original RS code in AWGN channel.

2.4.2 Power Consumption Analysis

In the previous section, we evaluated and compared the performance of both decoding algorithms as well as the un-coded data using the BER and SER as metrics. Next, we switch gears to evaluate power consumption for each method. The conditional BER figures will be used throughout this section to estimate the power consumption. It is important to highlight that the following calculations are based on the AWGN model using RS(15,9)/MRS(15,9) code. Similar evaluation methodology applies to other RS codes in both channel models. We left the details for the reader and summarized the energy consumption and saving in Table 2.5 and 2.6.

We start by defining our system setup that has a moderate BER value of 10^{-4}, low throughput of 10k bits/sec, thermal noise of -131dBm, and receiver noise figure of 3 dB. Also assume that sensor nodes are separated by distance d of 10 m and experience a path loss exponent of 3.5. Defining the system parameters is essential in order to estimate the power consumption using the equations in Section 2.3. From Fig. 2.8, the required SNR to achieve BER of 10^{-4} is estimated at 11 dB for the un-coded data. Then, the power consumed in transmitting the bits of the un-coded data can be calculated by using equation 2.14 and

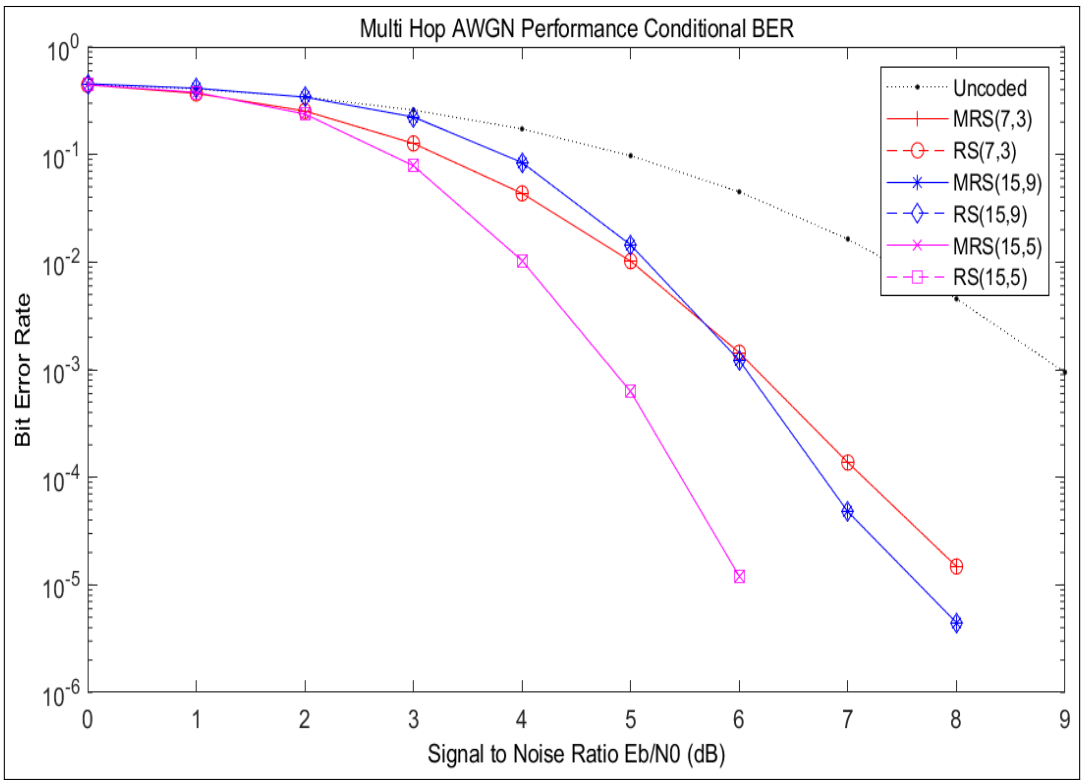

Figure 2.8: Conditional bit error rate for un-coded data, modified RS code, and original RS code in AWGN channel.

2.15. After substituting these values into 2.14 and 2.15, P_{tx} is estimated at -35.58 dBm and transmission energy per bit E_{tx} is estimated at 2.77 nJ/bit.

To calculate P_{tx} for RS(15,9)/MRS(15,9) code, we first examine Fig. 2.8 to determine G_{ECC}. Since we do not require the data to the right of Th to be correct[1], the conditional BER curve is used to estimate P_{tx}. RS(15,9)/MRS(15,9) codes achieve G_{ECC} of 4.2 if used instead of the un-coded data in the standard system. P_{tx} and E_{tx} are estimated at -39.78 dBm and 1.05 nJ/bit, respectively. Now, we need to figure out how much energy the original RS decoding algorithm requires to decode the data. Since no actual sensor nodes were used, we estimated E_{rx} at 0.42 nJ/bit using [15, 16]. To calculate the energy saving per bit (ΔE) that we gain using ECC, we need to subtract the sum of E_{tx} and E_{rx} of RS(15,9) code from E_{tx} of the un-coded data. The energy saving per bit is estimated at 1.30 nJ/bit.

At this point, we need to analyze the energy saving gain resulted from using the modified RS decoding algorithm over the original one. The RS decoding algorithm profiled base on the time each component block of the algorithm spend on computation. The profiled algorithm is detailed in Table 2.4. It is worth mentioning, that our proposed Modified algorithm bypass

[1]recall conditioning the BER and SER on the correctness of bits/symbols to the left of Th yields to equivalent performance from both RS decoding algorithms

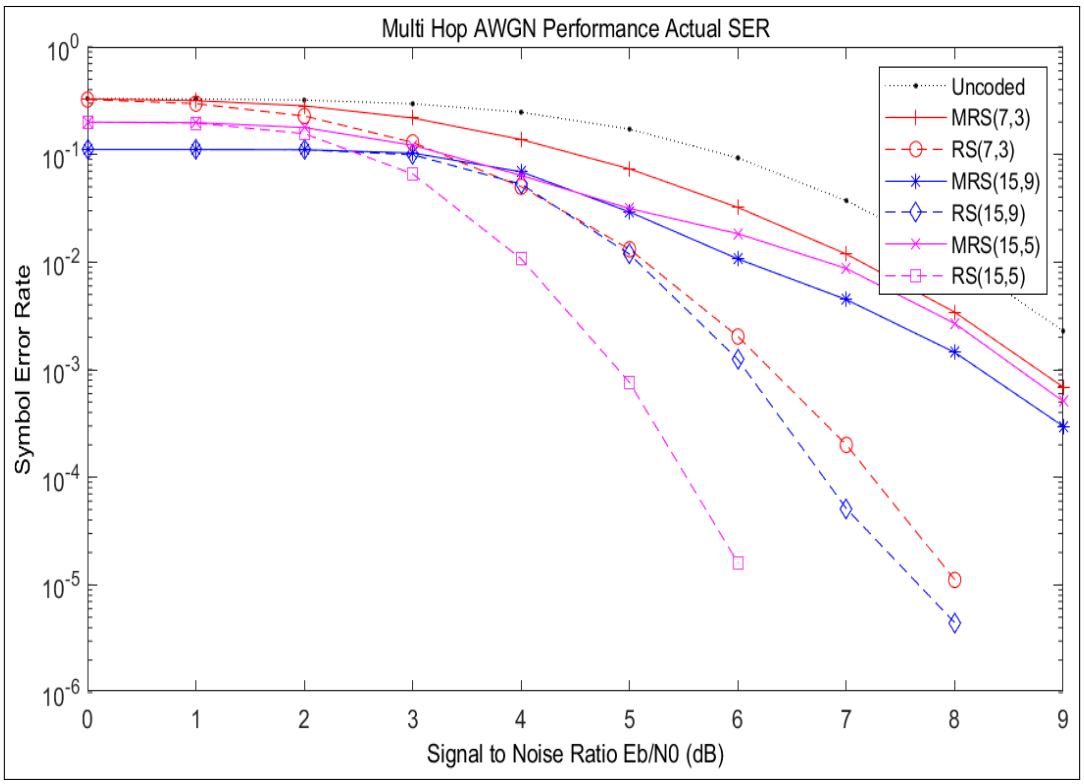

Figure 2.9: Actual symbol error rate for un-coded data, modified RS code, and original RS code in AWGN channel.

the last two steps namely the error evaluator and the error corrector which represents 15% of the total processing time. Using MATLAB, we found that after transmitting 1,800,000 code words, 563,148 correctable code words were received. Out of these 563,148 code words, 197,451 perceived as correct because all errors are located to the right of Th. This comprises 35% of the correctable code words that were partially processed by the modified RS decoding algorithm. Since the modified algorithm still must find the location of errors in the data but skips evaluating the magnitude of these errors, this reduces the average E_{rx} from 0.42 nJ/bit to 0.40 nJ/bit and results in energy saving per bit of 1.32 nJ/bit instead of 1.30 nJ/bit on average.

Table 2.5 compiles the energy consumption for each RS code in the multi-hop AWGN channel case. The first column reflects the gain obtained by using a specific RS code instead of the un-coded data. The second and the forth columns reflect E_{tx} and E_{rx}, respectively. In the third column, the percentage of code words perceived as correct by the modified RS decoding algorithm is presented. Finally, the fifth column shows the energy saving per bit using a specific algorithm compared to the un-coded data. From the table, it is obvious that the modified RS decoding algorithm consumes less energy in comparison to the original RS decoding algorithm. Also, we can observe that MRS(15,5) and MRS(7,3) achieve higher

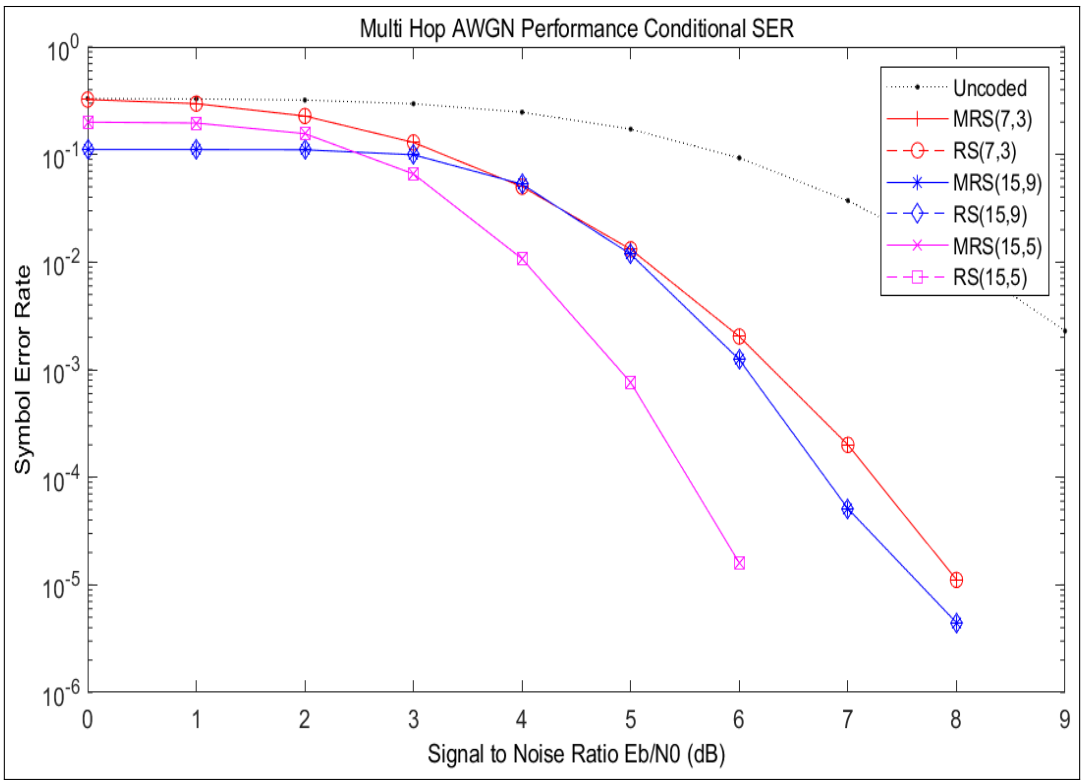

Figure 2.10: Conditional symbol error rate for un-coded data, modified RS code, and original RS code in AWGN channel.

reduction percentage in the decoding energy in comparison to MRS(15,9). This is because the number of information symbols is comparable to error correcting capabilities of these codes. In addition to that, the modified RS decoding algorithm ignores errors when they only occur in the parity symbols. Similarly, we can justify the higher percentage achieved by MRS(7,3) in comparison to MRS(15,5) using the same exact reasons. At last, it is worthy to mention that these results are obtained using the previously mentioned system setup.

Similar calculations can also be obtained for the Rayleigh fading model using RS(15,9). However, we will limit our power consumption analyses in the Rayleigh fading model to only discuss the energy savings in the decoder. As in AWGN model, after transmitting 1,800,000 code words, 318,520 correctable code words were received. Out of these 318,520 code words, 85,452 perceived as correct because all errors are located to the right of Th. This comprises 27% of the correctable code words that were partially processed by the modified RS code. Using a similar argument to the one in the previous paragraph this reduces the average E_{rx} from 0.42 nJ/bit to 0.36 nJ/bit.

Table 2.6 similar to Table 2.5 shows the decoding energy saving as a result of using the modified decoding algorithm over the original one. We can observe that MRS(15,5) and MRS(7,3) achieve higher reduction percentage in the decoding energy in comparison

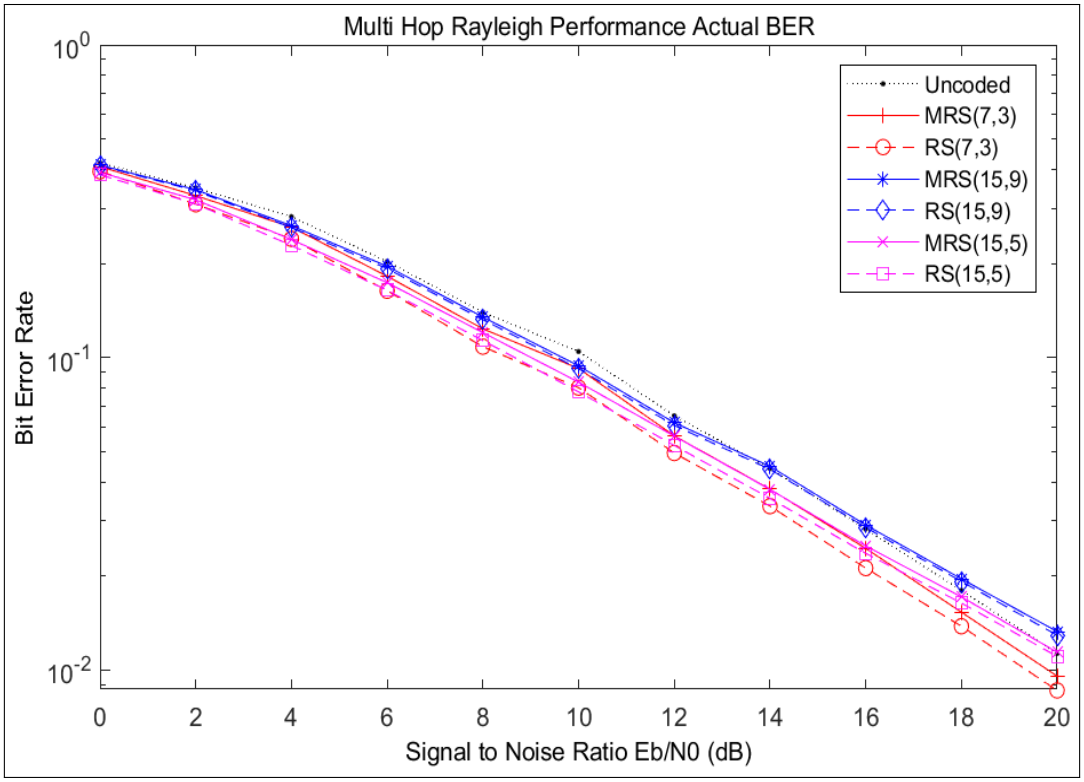

Figure 2.11: Actual bit error rate for un-coded data, modified RS code, and original RS code in Rayleigh channel.

to MRS(15,9). Again, this is because the number of information symbols is comparable to error correcting capabilities of these codes. In addition to that, the modified RS decoding algorithm ignores errors when they only occur in the parity symbols. Clearly, we can note that MRS(15,5) outperforms both MRS(15,9) and MRS(7,3) since MRS(15,5) has the more parity symbols in comparison. Also, MRS(15,5) has a higher capability in correcting errors compared to the MRS(15,9) and MRS(7,3). Furthermore, it is worthy to note that reduction in energy under AWGN channel is higher than Rayleigh channel which caused by the severe degradation in the communicated data.

2.5 Conclusion

In this work, we evaluated the performance and power consumption of the modified RS decoding algorithm, the original RS decoding algorithm, and the un-coded data in a multi-hop AWGN and Rayleigh fading channels using multiple RS codes. The modified RS decoding algorithm provides the required data protection to the bits/symbols located to the left of Th while maintaining a good performance, and it reduces the total power consumed in the decoder. In the Rayleigh fading channel, the modified decoding algorithm performs close to

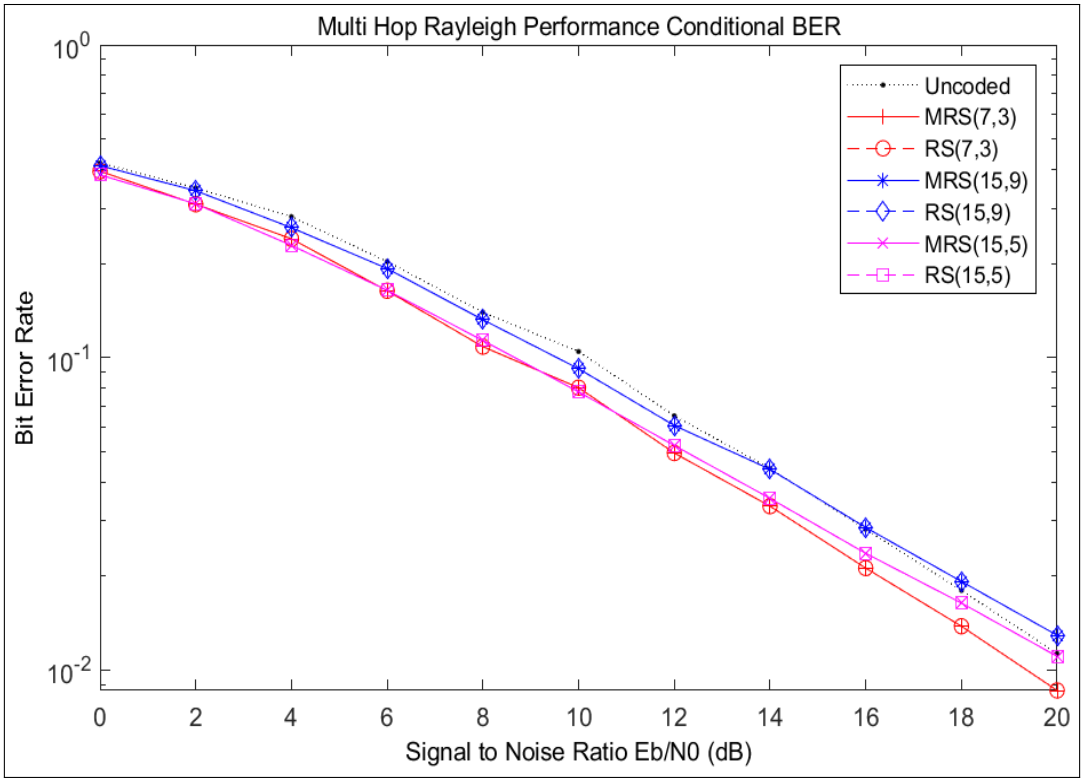

Figure 2.12: Conditional bit error rate for un-coded data, modified RS code, and original RS code in Rayleigh channel.

the original RS code. For example, the modified RS decoder reduces the average consumed energy per bit on average by 7% in the AWGN case and by 6% in the Rayleigh fading case. Though the modified version of RS decoding algorithm is able to save energy, it is clear that it is not enough to justify the loss in performance MRS suffer compared to the original. However, developing algorithm that consumes less energy to evaluate and determine error locations will increase the overall energy savings [25].

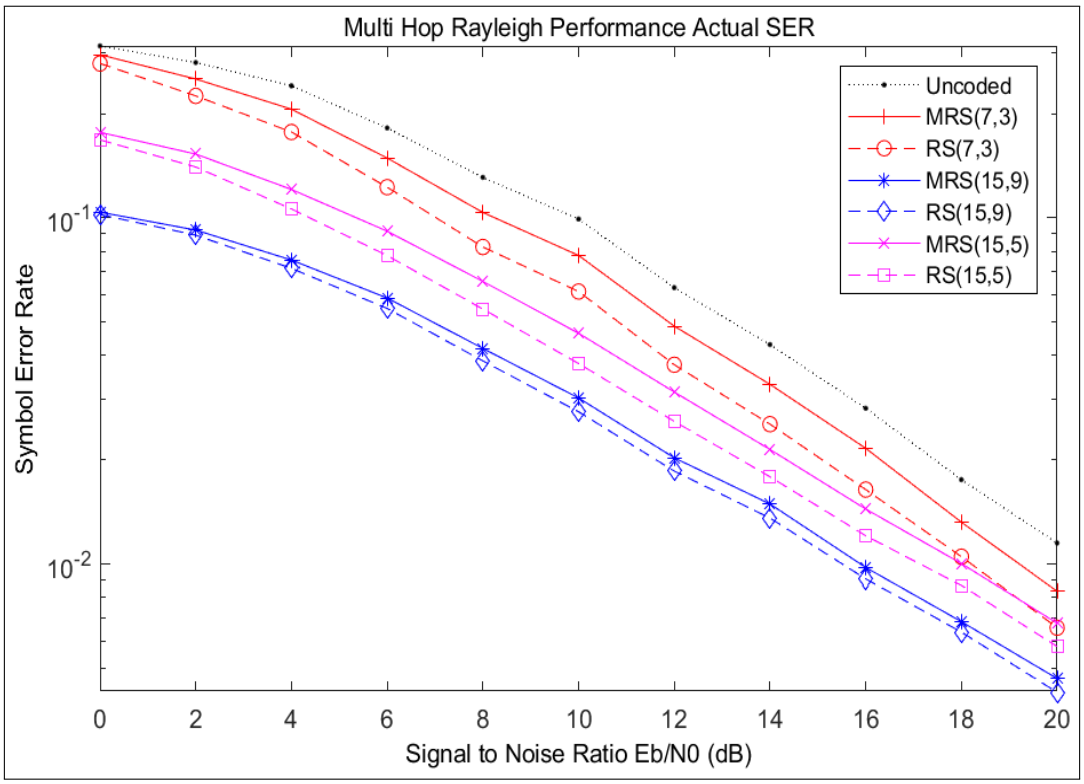

Figure 2.13: Actual symbol error rate for un-coded data, modified RS code, and original RS code in Rayleigh channel.

Table 2.4:
Time the processor spends per Reed-Solomon block

Block	Percentage of time
Syndrome calculation	55%
Error locator	6%
Chien search	23%
Error evaluator	12%
Error corrector	3%

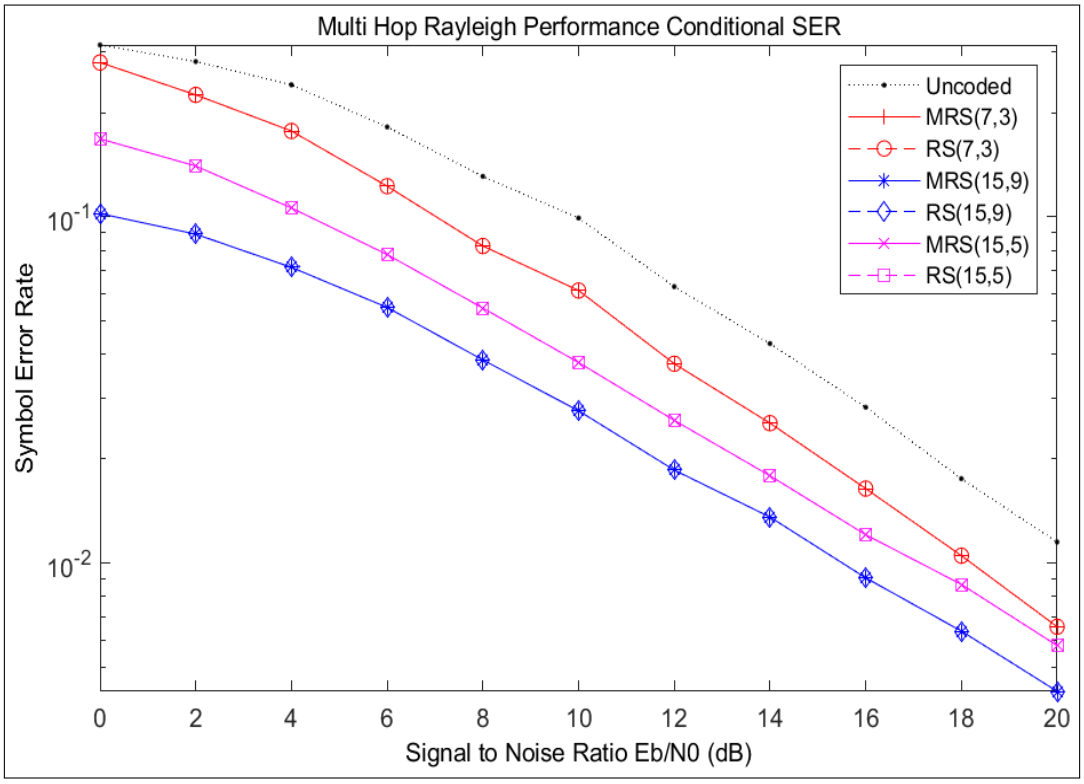

Figure 2.14: Conditional symbol error rate for un-coded data, modified RS code, and original RS code in Rayleigh channel.

Table 2.5:
Energy consumption of RS codes in a multi-hop AWGN channel

	G_{ECC} (db)	E_{tx} (nJ/bit)	% of reduction in decoder energy	E_{rx} (nJ/bit)	ΔE (nJ/bit)
RS(7,3)	3.8	1.15	—	0.42	1.20
MRS(7,3)	3.8	1.15	8.3%	0.39	1.23
RS(15,9)	4.2	1.05	—	0.42	1.30
MRS(15,9)	4.2	1.05	5.5%	0.40	1.32
RS(15,5)	5.5	0.78	—	0.42	1.57
MRS(15,5)	5.5	0.78	7.5%	0.39	1.60

Table 2.6:
Decoder energy consumption of RS codes in a multi-hop Rayleigh channel

	% of reduction in decoder energy	E_{rx} (nJ/bit)
RS(7,3)	—	0.42
MRS(7,3)	6%	0.40
RS(15,9)	—	0.42
MRS(15,9)	4.5%	0.40
RS(15,5)	—	0.42
MRS(15,5)	7.5%	0.39

Chapter 3: Diversity Network Coding with Cooperation

Network coding (NC) represents a fundamental change in approach, yet simple idea, in packet networks. Instead of the simple store-and-forward mechanism, network coding allows intermediate nodes to combine and compute functions of received packets before passing them toward their destination. NC first introduced in the influential paper of Ahlswede et al. in [26] which demonstrated the advantages of their proposal. Since its inception, network coding proved its capabilities to improve transmission efficiency, throughput, and delay over broadcast channels.

In wireless systems, two NC design strategies highlighted in literature namely, random network coding and opportunistic network coding. In random network coding, intermediate nodes combine all source packets using random and independent coefficient. While opportunistic network coding exploits the diversity of lost and received packets at each intermediate node to achieve a certain goal. Although, random network coding has its advantages such as the ability to recover packets without feedback and reducing number of packet transmissions, it is only achievable in applications with high delay tolerance. Additionally, it is inadequate in unicast and multicast settings where different receivers are concerned in diverse subsets of the transmitted packets. Therefore, in this research we decided to work with opportunistic network coding since it addresses the previously highlighted concerns, however, it comes with its own drawbacks such as scalability issues and sub-optimal throughput in comparison to random network coding. Table 3.1 [27], shows a detailed comparison between different type of network coding and highlight the strength and weakness of each method.

Network coding with cooperation implementation has recently grown due to the potential improvement in terms of diversity order and throughput in comparison to conventional techniques. Intermediate nodes linearly combine multiple input packets and then forward it to the destination or other intermediate nodes in the network. Therefore, the intermediate nodes can serve multiple sources in a single time slot which correspond closely to how wireless sensor networks behave. Cooperation diversity enables nodes to exchange packets required by other nodes to successfully decode network coded packets when needed. Moreover, cooperation diversity helps in reducing the bit error rate as well as decentralizes the transmitted power among all nodes in the network.

Diversity coding was introduced a decade before network coding [28, 29], it was proposed as protection mechanism for link failure. Diversity coding shares the same principles with network coding. Initially the main goal was to provide an additional coded link beside the

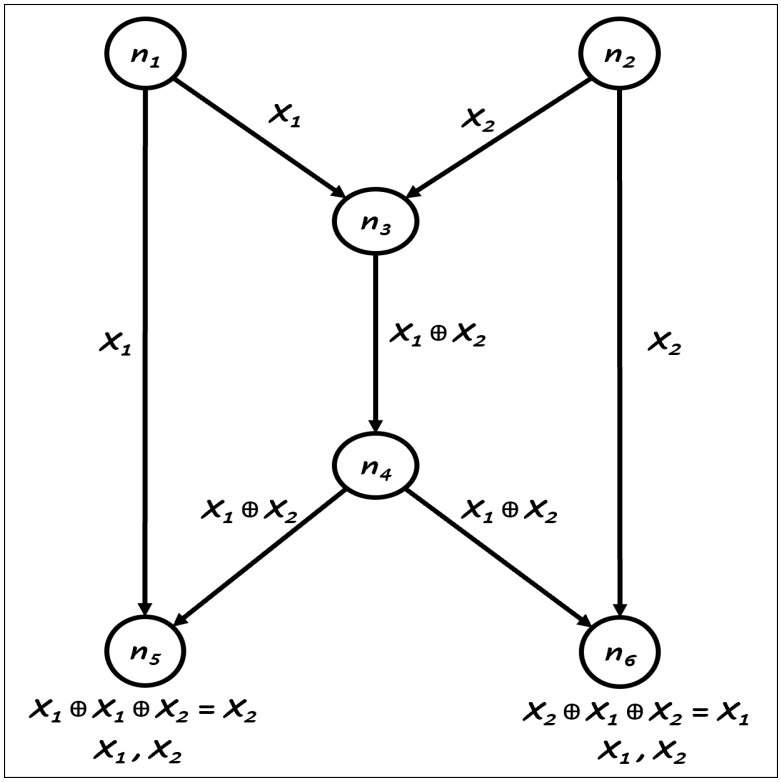

Figure 3.1: Butterfly network model showing network coding.

actual data links to support near-instantaneous recovery in case of link failure, however, diversity coding found its applications in many research proposals [30–32]. We deploy diversity coding to diversify the coded packets in the wireless sensor network to increase the network reliability and enhance its ability to recover data packets successfully.

Although the aforementioned topics has been widely investigated in literature, the use of error correction techniques with opportunistic network coding and cooperation has not been explored. Therefore, we decided to investigate the implementation of diversity network coding with cooperation to mitigate the challenges in implementing network coding in real time network specifically in wireless ad-hoc networks. We examine the performance of the proposed mechanism in static and mobile models and show that proposed solution provide superior performance when compared to direct transmission and opportunistic network coding.

3.1 Background

3.1.1 Network Coding

Network coding core principle is to motivate intermediate nodes to combine multiple data packets to reduce the number of transmissions required to deliver these packets to their

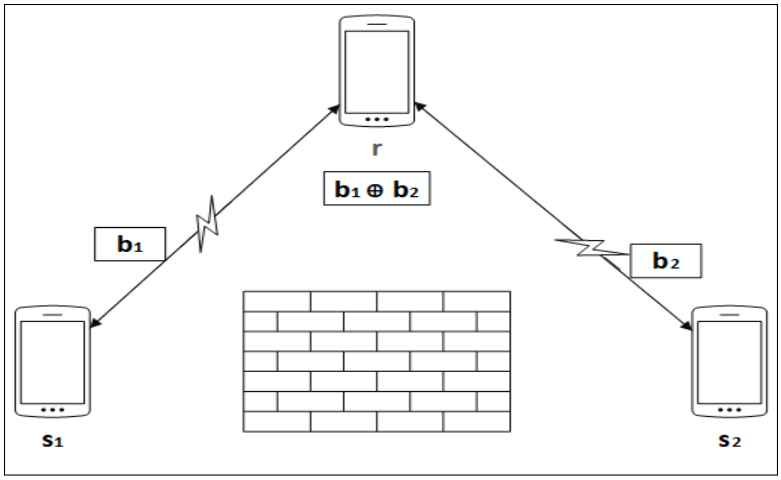

Figure 3.2: Wireless butterfly network model showing network coding.

destination(s). It is based on the simple butterfly network model illustrated in Fig. 3.1. In this model, a single source wants to multicast packets to two sink nodes. Each directed link represents an error free channel which can deliver the packet in single transmission to the next node. Here network coding, achieves an improved throughput of two packets per channel use. The source nodes first transmit two packets x_1 and x_2, but rather than transmitting one at a time at n_3, node n_3 transmits the module-two sum $x_1 \oplus x_2$. At node n_5 and n_6, they receive x_1 and $(x_1 \oplus x_2)$, and x_2 and $(x_1 \oplus x_1)$ respectively. Finally, since $x_1 \oplus (x_1 \oplus x_2) = x_2$ and $x_2 \oplus (x_1 \oplus x_2) = x_1$ each destination node can recover x_1 and x_2 successfully.

In wireless networks, the butterfly network model is depicted as in Fig. 3.2. As shown in the figure, s_1 and s_2 can not communicate directly with each other but they have to communicate through the relay node r. In this case, s_1 and s_2 transmits its packet to r and then r combine the packets and relay them to both nodes.

Table 3.1:
Performance of random, opportunistic, and instantly decodable network coding according to various criteria

Criterion \ Scheme	Random Network Coding	Opportunistic Network Coding	Instantly Decodable Network Coding
Throughput	Optimal	Sub-optimal	Sub-optimal
Delay	Huge delay	Moderate depending on the scheme	Moderate depending on the scheme
Complexity	Large field size	Depends on the scheme	Binary field
Encoding	Mix using random independent coefficients	Mix using diversity of lost and received packets	Mix using binary XOR
Decoding	Complexity cubical with the number of packets	Moderate depending on the scheme	Simple binary XOR
Progressive Decoding	Decoding is performed after getting the whole frame	Depend on the scheme but usually better than RNC	Instantaneous decoding
Overhead	Moderate depending on the scheme	Moderate depending on the scheme	Minimal
Buffer Size	As large as the frame size	Moderate depending on the scheme	No need for buffer
Feedback Load	Minimal feedback and can run even without feedback	More or less heavy depending on the scheme	Performance heavily depends on feedback
Broadcast Efficiency	Optimal	Sub-optimal	Sub-optimal
Multicast Efficiency	Inefficient	Depends on the scheme	Depends on the scheme

3.1.2 Opportunistic Network Coding

COPE protocol is the first attempt of implementing a practical opportunistic network coding in wireless networks [33]. The aim of this protocol is to increase network throughput using the current existing network stack by integrating network coding [33, 34]. Additionally, COPE takes advantage of the wireless broadcast nature and uses the packets available to encode and decode its network coded packets. COPE protocol consists of three components.

1. **Opportunistic Listening** COPE requires nodes to listen and store packets to decode network coded packets. These packets are of two types: (a) Packets that nodes broadcast themselves. (b) Packets that nodes overheard.

2. **Opportunistic Coding** COPE aims to maximize the number of packets combined at relay nodes while guaranteeing that each intended next-hop/ destination can decode its packet. To ensure that a simple rule is followed. To transmit k packets to k relay/destination nodes, a node only can XOR k packets if and only if each receiving node r_i has all $k-1$ packets x_j for $j \neq i$. This rule guarantees that each node receives a combined packet can decode it and extricates its packet.

3. **Reception Reports and Guessing** COPE implements a reception report that each node transmits to its neighbors. The reception report is attached to the packets the node transmits, otherwise, if the node has no packets to send it transmits its report through special control packets. Additionally, COPE leverages the wireless routing protocol which computes the delivery probability between every pair of nodes. It uses this information to guess if a neighboring node has a particular packet in the transmitting node forwarding queue.

3.1.3 Diversity Coding

Diversity coding was introduced as link failure recovery mechanism in [28, 29]. Though it shares the principles of network coding, diversity coding predates the introduction of network coding. The initial idea, assumes that there are multiple data streams being transmitted over multiple disjoint paths. Then, these links are protected against single link failure by a coded stream also known as parity link as follows:

$$c = x_1 \oplus x_2 \oplus \cdots \oplus x_n = \bigoplus_{i=1}^{n} x_i \tag{3.1}$$

Then if one of the data streams fails, the receiver can retrieve the failed link as follow:

$$c \oplus \bigoplus_{\substack{i=1 \\ i \neq j}}^{n} x_i = x_j \oplus \bigoplus_{\substack{i=1 \\ i \neq j}}^{n} (x_i \oplus x_i) = x_j \tag{3.2}$$

It is clear here that the recovery is near-instantaneous and accomplished without feedback or re-transmission. The basic idea of diversity coding is depicted in Fig. 3.3. This scenario can be extended into multi-point to multi-point networks as described in [28, 29].

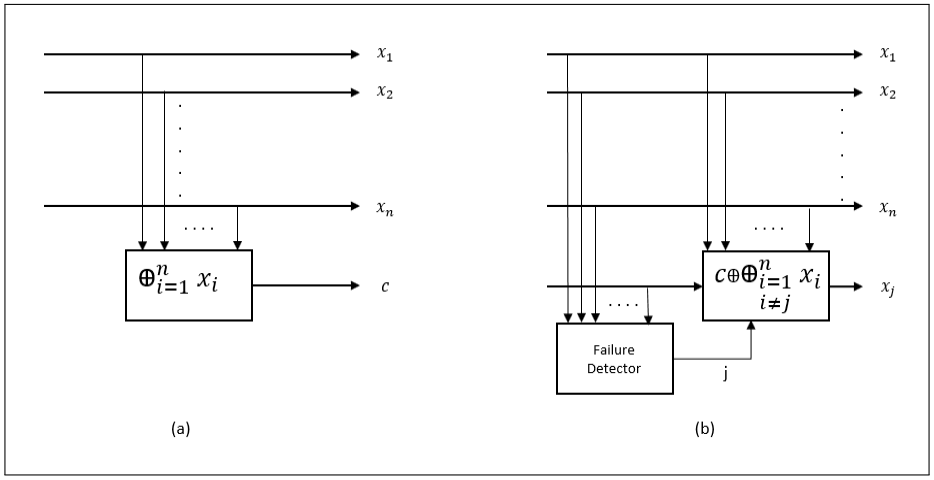

Figure 3.3: Diversity coding basic idea (a) encoder, (b) decoder.

3.1.4 Cooperative Communication

Cooperative communication is an influential technique to combat signal fading due to multi-path propagation in wireless networks. This concept was first introduced in [35–37] with basic idea that devices in close proximity can exchange packets to achieve a common or/and individual goal. It has shown its ability to tackle various challenges in the wireless network ranging from energy saving, throughput improvement, as well as reducing the overhead of recovering packets. It improves reliability by allowing destination nodes to combine packets received from multiple relays and better retrieve the original transmitted data. As long as destination receives adequate number of coded correct packets it will be able to decode them successfully. User cooperation has been widely implemented in many wireless network studies including network coding. It has been shown in [38] that user cooperation improves the performance of network coding. Therefore, we opted to incorporate in our design.

3.2 Proposed Protocol

In traditional Network Coding if any of the destination nodes fails to retrieve one of the source generated data packet due to movement or harsh channel conditions, it will fail to decode the XORed packet and extract the packets. As shown in Fig. 3.4 below if the link between node n_1 and n_5 fails, that means node n_5 will not be able to receive x_1 and eventually will not be able to extract x_2 from the XORed packet. However, if we combined diversity

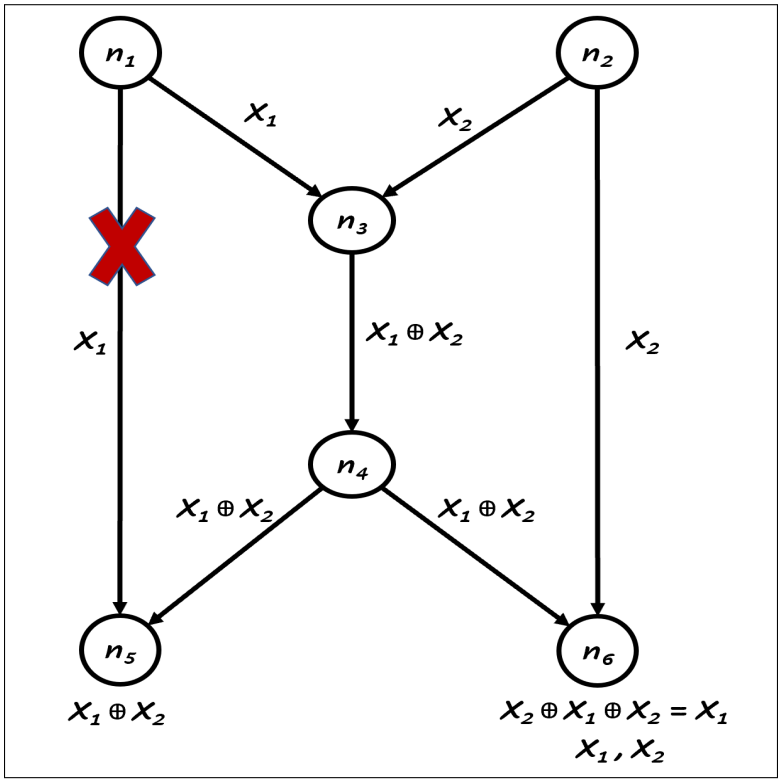

Figure 3.4: NC link failure.

coding with network coding, we can increase link reliability and ensure destinations nodes can retrieve the packets even if one of the links failed. By increasing the number of coded packets it ensure the ability of destination nodes to retrieve data. It is clear by increasing the number of redundant packets destination nodes do not have to relay on a single relay or source node to successfully deliver its packets, eliminating a significant shortcoming of originally proposed network coding scheme. To illustrate the diversity and network coding let us examine the following Fig. 3.5. In this scenario, node n_1 and node n_2 sends their packets to nodes n_3,n_6 and node n_3,n_7 respectively. At node n_3, it encodes the incoming packets and send the encoded packets to node n_4 and n_5 as follows:

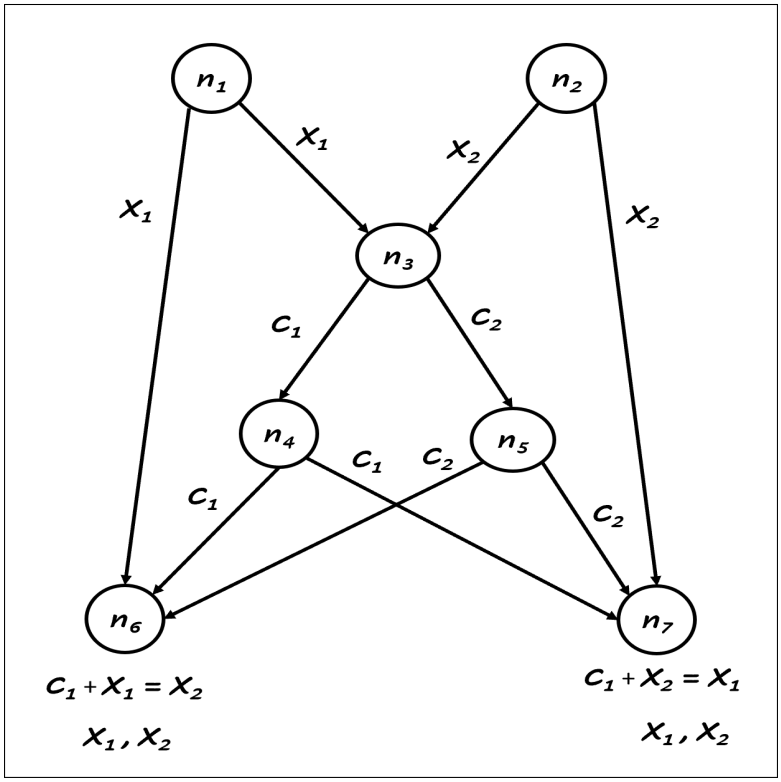

Figure 3.5: Diversity network coding.

$$\begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} \beta_{11} & \beta_{12} \\ \beta_{21} & \beta_{22} \end{bmatrix}^T \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \tag{3.3}$$

This results in the two network coded packets described in 3.4

$$\begin{aligned} c_1 &= \beta_{11}x_1 + \beta_{21}x_2 \\ c_2 &= \beta_{12}x_1 + \beta_{22}x_2 \end{aligned} \tag{3.4}$$

Where $\beta_{11}, \beta_{12}, \beta_{21}, \beta_{22}$ are the parity generator matrix rows for c_1, c_2. The β coefficients can be randomly selected, however, linear independence is not guaranteed and it depends on the GF field being used. Alternatively, the β coefficients can be selected from Vandermonde matrix making these coefficients known to all the network nodes and assure their linear independence. The β_{ij} coefficients are calculated as follow:

$$\beta_{ij} = \alpha^{(i-1)(j-1)} \tag{3.5}$$

where α is a primitive elements of GF(2^q), indices $i \in \{1, 2, \cdots, k\}$ and $j \in \{1, 2, \cdots, k\prime\}$, and k is the number of packets at the relay node and $k\prime$ is the number of coded packets generated

by relay node and $k\prime \geq k$. In this work the case $k\prime = k$ is considered. The coefficient matrix is defined as:

$$B = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \alpha & \alpha^2 & \cdots & \alpha^{(k\prime-1)} \\ 1 & \alpha^2 & \alpha^4 & \cdots & \alpha^{2(k\prime-1)} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 1 & \alpha^{(k-1)} & \alpha^{(k-1)2} & \cdots & \alpha^{(k-1)(k\prime-1)} \end{bmatrix} \tag{3.6}$$

Node n_6 and n_7 can extract the packets by using x_1, c_1 and x_2, c_1, respectively. If there was no link failure at any node, destination nodes n_6 and n_7 can ignore the coded packet c_2. Node n_6 can decode for both x_1 and x_2 as follow:

$$\begin{aligned} \tilde{c_1} &= c_1 + \beta_{11}x_1 \\ \tilde{c_1} &= \beta_{11}x_1 + \beta_{21}x_2 + \beta_{11}x_1 = \beta_{21}x_2 \\ x_2 &= \frac{\tilde{c_1}}{\beta_{21}} \end{aligned} \tag{3.7}$$

where $\tilde{c_1}$ is the combined result of the coded packet c_1 and the packet x_1 after multiplying it with β_{11} coefficient.

In case of a link failure as shown below. Node n_6 can use c_1 and c_2 to retrieve x_1 and x_2 as follows:

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} \beta_{11} & \beta_{21} \\ \beta_{12} & \beta_{22} \end{bmatrix}^{-1} \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} \tag{3.8}$$

Alternatively, if the link between node n_4 and n_6 fails then node n_6 can still retrieve x_2 by using c_2 as follows:

$$\begin{aligned} \tilde{c_2} &= c_2 + \beta_{12}x_1 \\ \tilde{c_2} &= \beta_{12}x_1 + \beta_{22}x_2 + \beta_{12}x_1 = \beta_{21}x_2 \\ x_2 &= \frac{\tilde{c_2}}{\beta_{22}} \end{aligned} \tag{3.9}$$

Moreover, the above framework can be extended to encode three packets. if a node attempts to encode 3 packets we need a 3x3 parity generator matrix to achieve diversity as shown below.

$$\begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} \beta_{11} & \beta_{12} & \beta_{13} \\ \beta_{21} & \beta_{22} & \beta_{23} \\ \beta_{31} & \beta_{32} & \beta_{33} \end{bmatrix}^T \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \tag{3.10}$$

To retrieve packets a similar procedure can be followed as detailed before. The details for

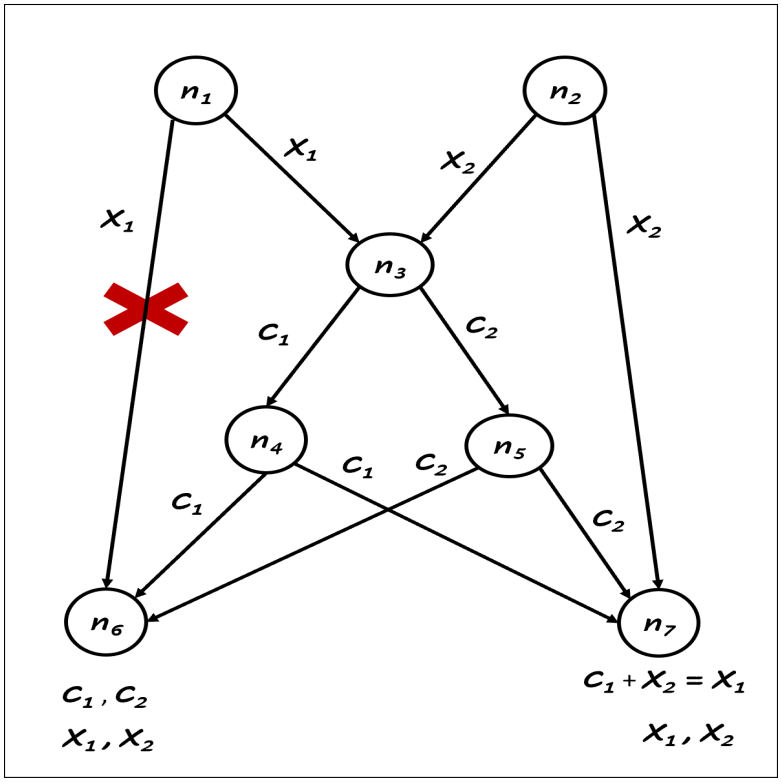

Figure 3.6: Diversity network coding with link failure.

it is left for the reader. In this work we will limit the maximum number of network coded packets to three. It has been shown in [39], that majority of network coded packets consists of two to three packets as shown in Table 3.2. Our algorithm has similar conditions to what is proposed in PNC-COOP and the majority of network coded packets consists of two or three packets. Therefore, it is reasonable to limit our work to maximum of three packets and the possibility to encode more than three is left for future work.

3.3 System Model and Setup

3.3.1 System Model

The system consists of a grid network where each node is placed at the exact distance from other nodes in the x, y plane. Packets are generated at source nodes and transmitted in multicast to their destination nodes. It is assumed that source and destination nodes are not in the same coverage area and multi-hop transmission is required to deliver the data using relay nodes. At relay nodes, if there is more than one packet in the queue waiting to be transmitted to their destinations, the relay node performs diversity and network coding as explained in the previous section. We will limit the number of network coded packets up to 3 packets and

Table 3.2: Percentage of number of packets network coded together using ONC.

Solution	SNR(dB)	Number of packets network coded together (%)					Total NC %
		None	2	3	4	5	
ONC	0	85.4458	12.6361	1.7442	0.1694	0.0045	14.55
	5	86.7403	12.2758	0.9512	0.0305	0.0022	13.26
	10	82.7403	16.2351	1.5583	0.0359	0.0000	17.83
	15	85..2653	13.4815	1.2376	0.0156	0.0000	14.73
	20	83.6562	15.0329	1.2908	0.0201	0.0000	16.34
	25	82.6232	15.8889	1.4203	0.0676	0.0000	17.38
	30	83.9147	13.8831	2.0442	0.1557	0.0023	16.09

explore increasing the limit in future work. The resulting packets will be forwarded to the next relay nodes. Since the system deploy opportunistic network coding and diversity coding, the system depends profoundly on opportunistic listening and reception protocol. Nodes in the network are allowed to collect packets that are transmitted in its proximity. It is worth mentioning, that only correct packets are stored at the nodes that deploy opportunistic listening. This helps these nodes to decode network coded packets if needed and reduce the error propagation caused by network coding. CRC-CCITT is implemented to detect errors and only packets with no errors are kept at the listening nodes.

Additionally, relay nodes only apply network coding on direct packets, i.e. it uses the packets that were directly transmitted to the relay node not the packets collected using opportunistic listening. Moreover, relay nodes do not deploy waiting strategies to increase number of packets in order to increase network coding rate. This will help reduce the search space for nodes that can decode the network coded packets and reduce the end-to-end delay that might results by deploying such strategies. Furthermore, relay nodes will utilize the reception reports received periodically from other nodes in the network to determine which packets to be network coded and to maximize destination(s) ability to decode the packets.

Assume our network is represented as a graph $G(n, E(n))$, where n nodes are placed uniformly and independently on a grid and $E(n)$ is the set of edges connecting the network nodes. Also, it is assumed each node n_i is connected to its neighboring nodes if and only if they are within its transmission range. To evaluate our proposed solution, the system is evaluated under two study cases as described later. Additionally, with probability $P_{out}(E)$ an edge will experience an outage. It is assumed in all simulations that only one edge can experience an outage. Subsequently, the network is evaluated under two conditions. In the first case, network is static with no outages/mobility, i.e. nodes are assumed to stay stationary and all links assumed to be available with no change in their conditions throughout

Table 3.3: System parameters.

Parameters	Setting
Offered Load	10 Mbps
Modulation	QPSK
Carrier Frequency	2.4 GHz
Distance between node	30m
Payload size	1024 bytes
DIFS	50 \mu sec
SIFS	10 \mu sec
RTS/CTS/ACK	1 \mu sec
CWmin	7
CWmax	255
aSlotTime	15 \mu sec

simulations. In the second case, an edge E connecting a source node with its destination is subject to experience an outage with probability $P_{out}(E)$. In both cases, system is evaluated using Rayleigh fading channel, it is assumed that nodes within network are stationary or move slowly. Moreover, the channel is assumed to be frequency non-selective, slowly varying fading channel. This means, that the signal bandwidth is much smaller than the coherence bandwidth.

3.3.2 System Setup

The parameters under which simulation results are obtained are summarized in the following Table 3.3. Additionally, it is assumed that nodes have the same coverage area and number of neighbors. Also, it is assumed that source nodes always need three hops to reach destination and the number of neighboring nodes is set to 6. Moreover, it assumed that there is no packet loss/collision during CTS or ACK.

3.4 Simulation Results and Analysis

As we had discussed in the previous section, the system is evaluated under two cases namely static and mobile, respectively. We explore the effect of link outages due to mobility by varying the outage probability, we simulated the mobile case with P_{out} set to 0.001, 0.01, and 0.1 during each study case of mobility to reflect its severity, i.e in each round of simulation we assumed a link might fail during each transmission with constant P_{out}. The system is

evaluated by calculating BER between source and destination and the system as whole. BER is defined as the percentage of bits lost or received in error divided by the number of bits transmitted. Additionally, the system is evaluated base on average throughput, defined as number of bits received correctly at the destination nodes. The results for each of the cases are discussed under their perspective sections.

3.4.1 Case 1: Static Model

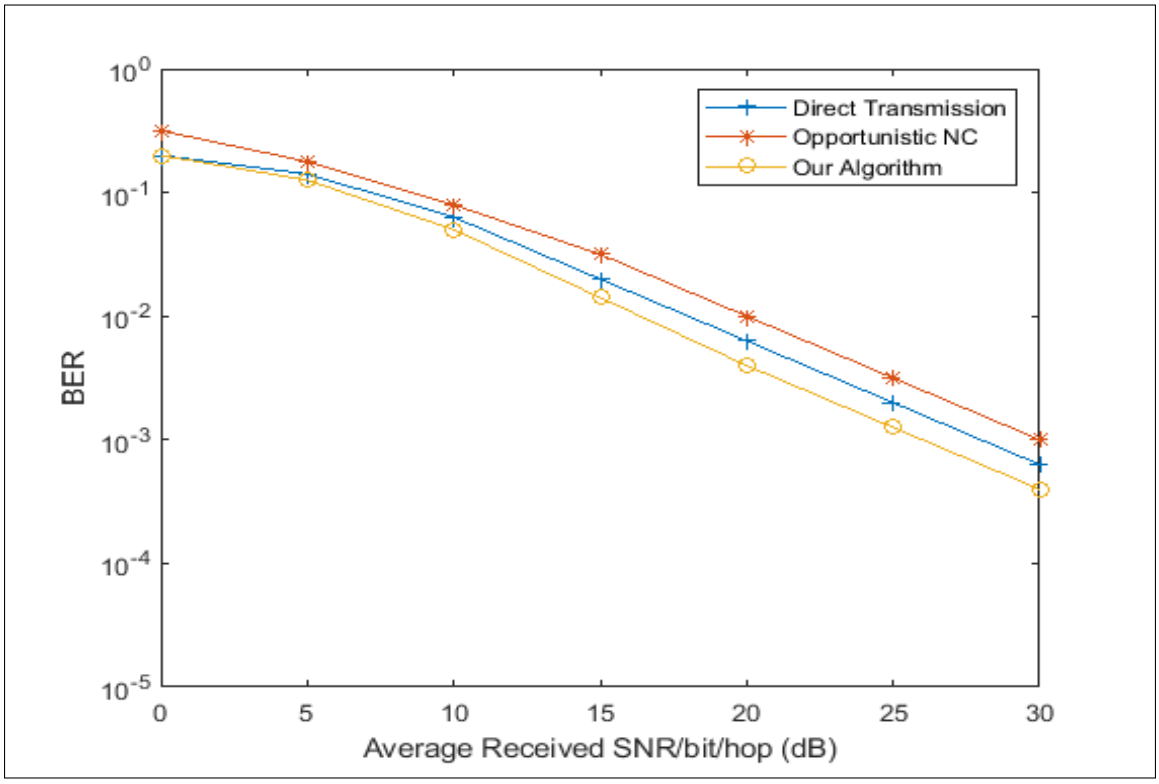

Figure 3.7: BER at specific destination with network coding capped at 2.

First, we start by evaluating the system when links and network nodes are static, i.e. no node link outages during transmission. We compare our proposed solution to opportunistic network coding (ONC) as well as the direct transmission (DTx). Fig. 3.7 and Fig. 3.8 shows the BER between a source and destination pair while capping the network coding capabilities to 2 and 3 packets, respectively. In both figures, it is clear that our solution provides a better performance in comparison to DTx and ONC. It outperformed ONC by 5dB and DTx by 3dB while achieving the same BER. Also, Fig. 3.9 shows the system overall BER, again our solution outperforms both DTx and ONC by same margin in the case of DTx and a substantial margin over ONC.

Theoretically, ONC should increase network throughput and deliver more data bit to

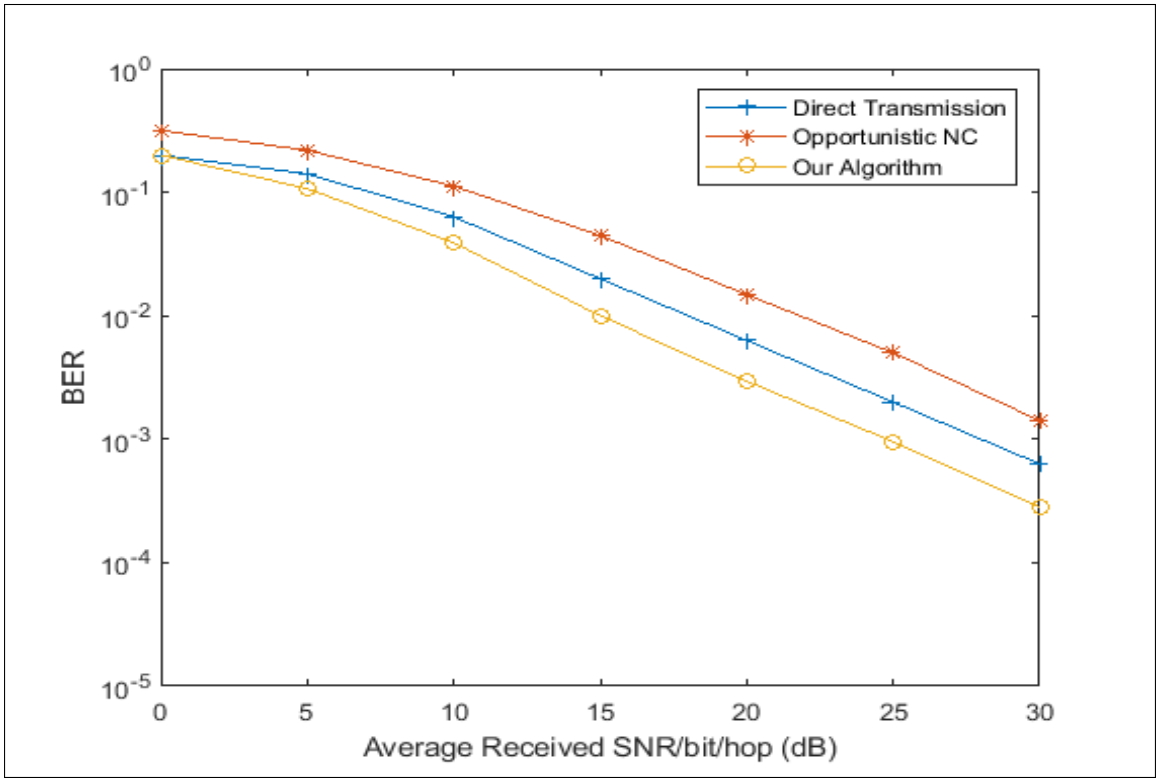

Figure 3.8: BER at specific destination with network coding capped at 3.

the destination nodes. Ideally, this is true, however, when implementing ONC in actual network with proper noise model it fails to achieve the expected performance. As depicted in Fig. 3.10, our solution provides a similar throughput performance in comparison to DTx and a significant improvement when it is compared to ONC. This caused by the fact that our proposed solution only encodes packets that are directly transmitted to relays and only encode packets that destination nodes are able to decode successfully. As well as the fact, that our solution limits the number of encoded packets to 3 packets. Subsequently, by limiting our solution to maximum of 3 packets we were able to mitigate some of the problems caused by mobility as we will see in the next section.

3.4.2 Case 2: Mobile Model

In this section we evaluate our solution when we take mobility/link outage under consideration. We consider three scenarios where a link might experience an outage with $P_o(E)$ of 0.001, 0.01, and 0.1 respectively. These probabilities will allow us to examine the performance with various network conditions. Fig. 3.11-Fig. 3.13, depicts the BER performance of our solution in comparison to DTx and ONC. It is clear from the figures that ONC suffer heavily with change in topology and experience a higher BER as network condition worsen. Also, it

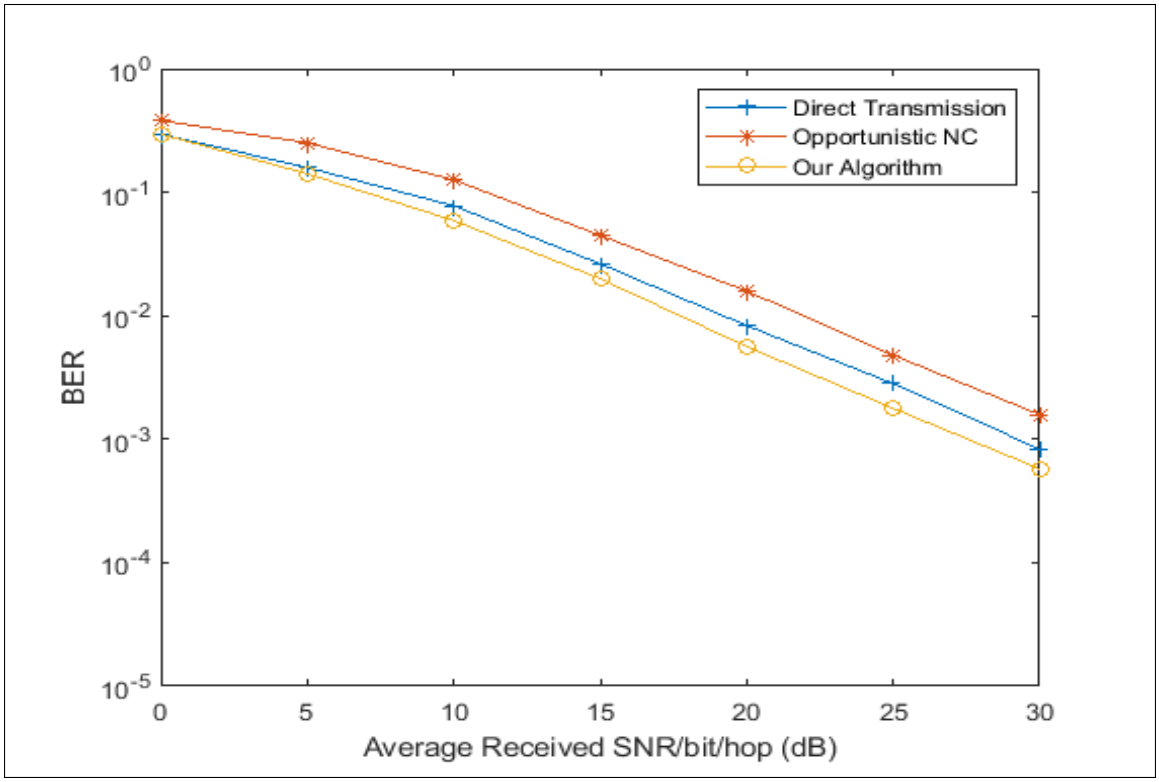

Figure 3.9: System BER.

can be concluded that DTx performs better and experiences less degradation in performance when compared to ONC. Moreover, we can see that our proposed solution maintains an acceptable performance as the network condition worsen and outperform both ONC and DTx as the network condition worsen. It is clear that our proposed solution offers more consistent performance in all scenarios and only experience a small degradation in performance as well.

Correspondingly, Fig. 3.14, and Fig. 3.15 depicts the network throughput when $P_o(E)$ is equal to 0.01, and 0.1, respectively. It is clear that ONC is under performing in comparison to DTx and our solution. Equivalently surprising, as the probability of links outages become more frequent, our solution starts to outperform the DTx. This expected as our solution diverse its packets and offer an alternative packets in which increases the ability of destination nodes to decode the packets.

3.5 Conclusion

In this work, we proposed a diversity network coding with cooperation. As opposed to network coding which encode packet by XORing the packets and forward them to their destinations making the decoding process reliant on receiving the XORed packet correctly. We

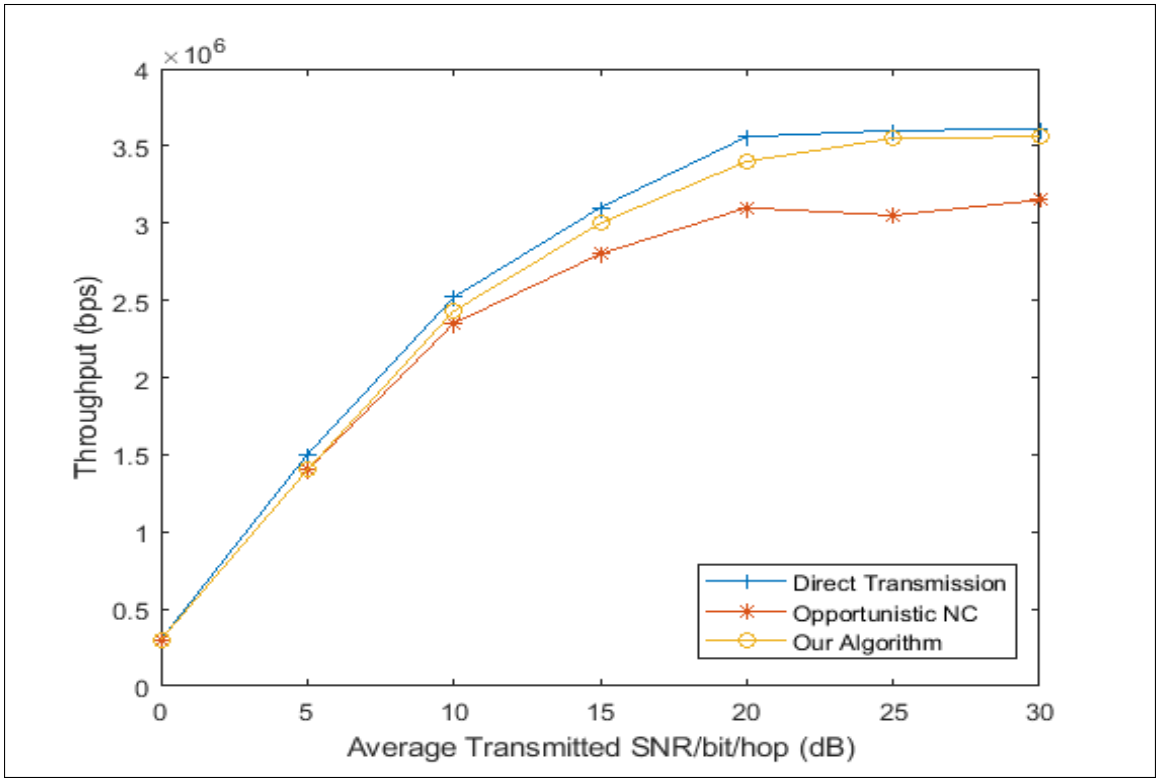

Figure 3.10: System throughput.

established redundancy by diversifying the network coded packets using linear combination. Creating alternative packets that can be used to retrieve the original packets in case one of the XORed packets was not deliver intact. Also, we proposed a no waiting strategy and limit the number of network coded packets to three. We argued that limiting the network coded packets to three is reasonable and we showed using simulation that the probability to encode more than three packets is insignificant. Deploying such techniques and limitations helped our solution to diminish the problems that affect NC when deployed in communication channel with an appropriate noise model.

We evaluated our proposed solution in two different network settings, static and mobile, respectively. We imitated the mobility network setting as link outage of one of its links, presenting a slowly moving nodes. We showed through simulation that our proposed mechanism provided a better BER performance in comparison to traditional NC or direct transmission. Moreover, we showed that our mechanism maintain a better throughput as the probability of link outage increases.

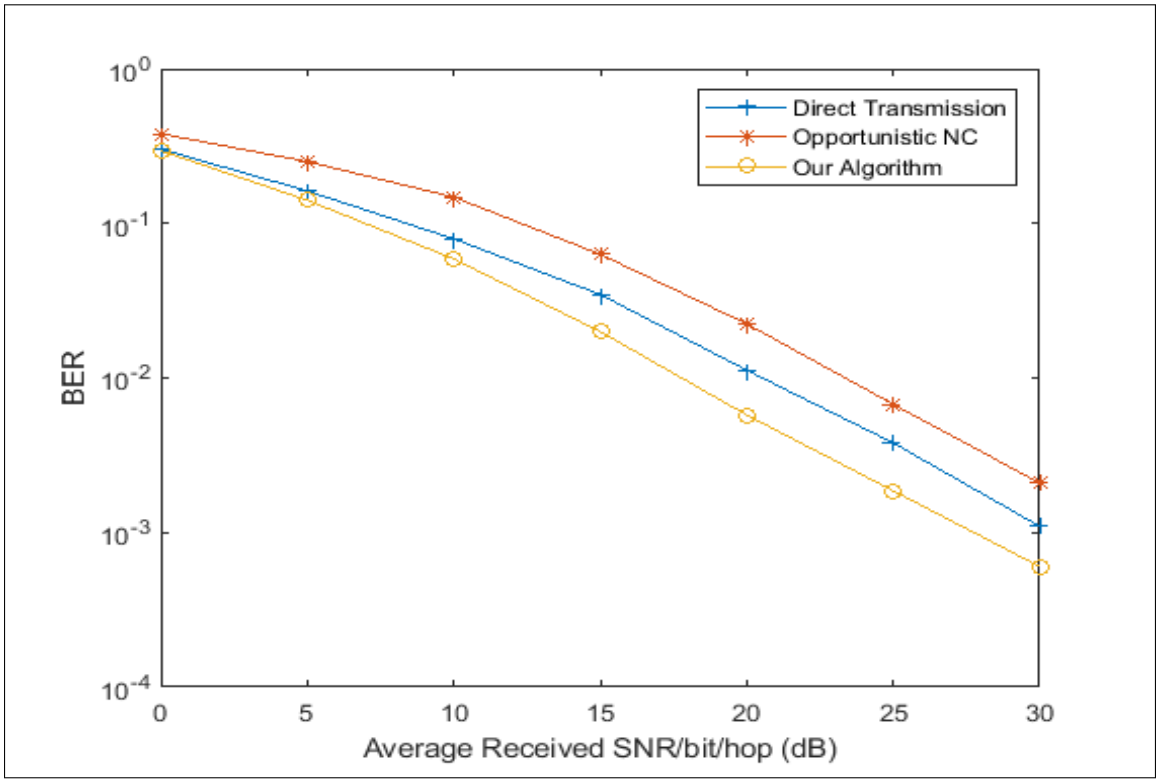

Figure 3.11: System BER $P_o(E)$=0.001.

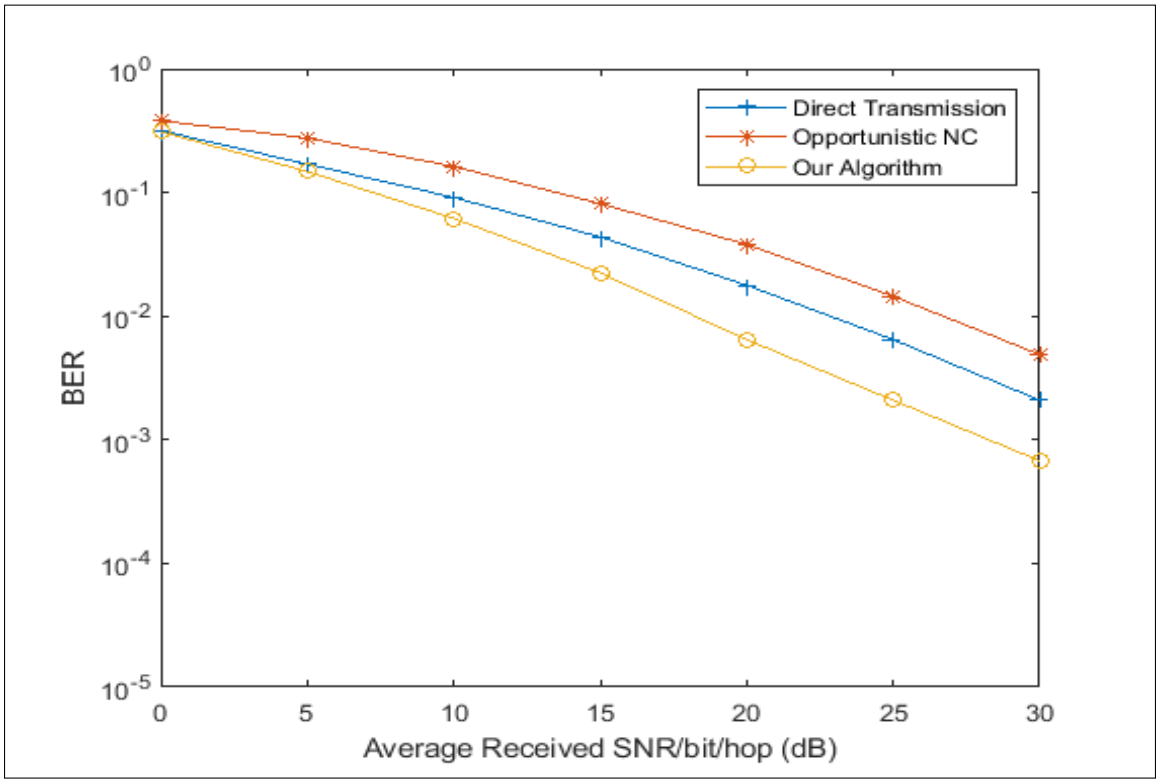

Figure 3.12: System BER $P_o(E)$=0.01.

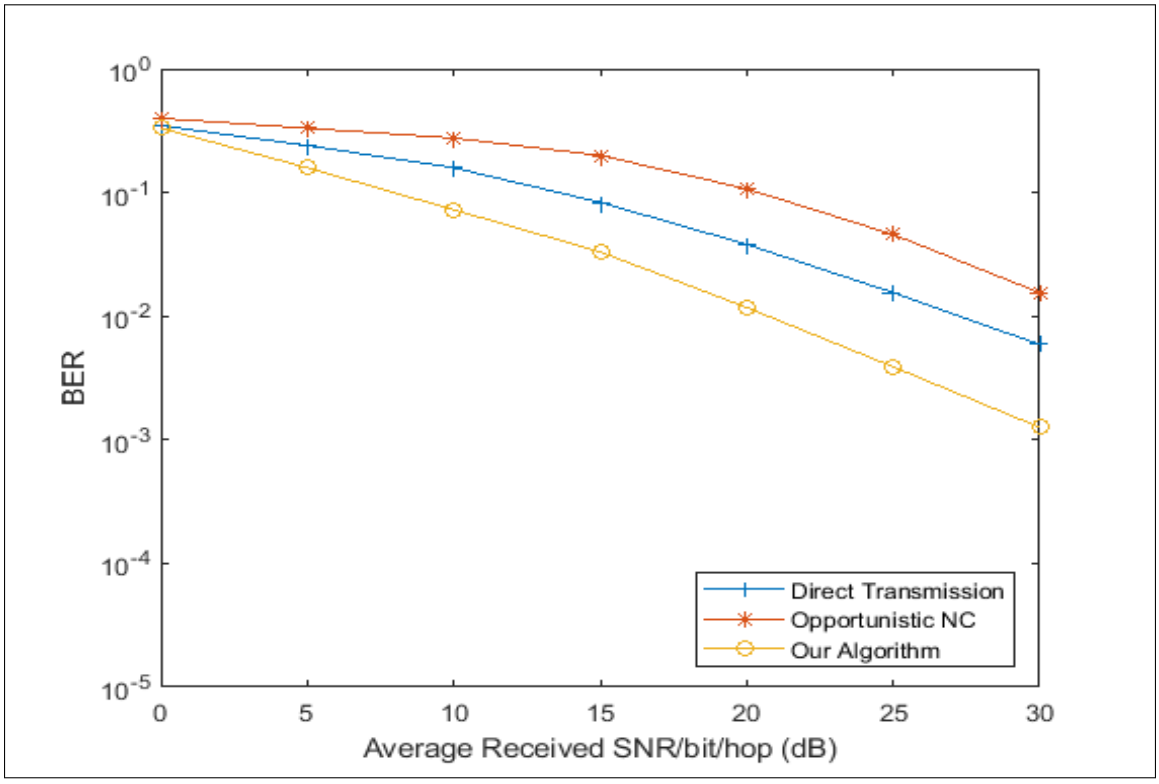

Figure 3.13: System BER $P_o(E)$=0.1.

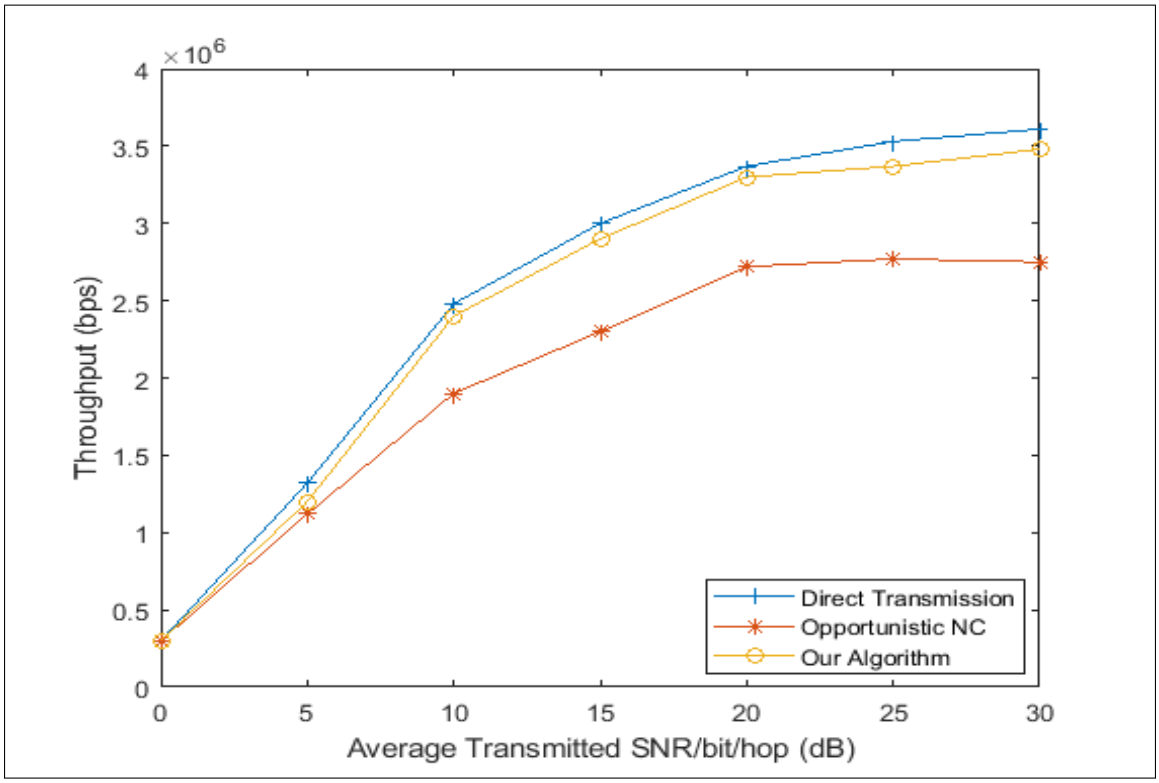

Figure 3.14: System throughput $P_o(E)$=0.01.

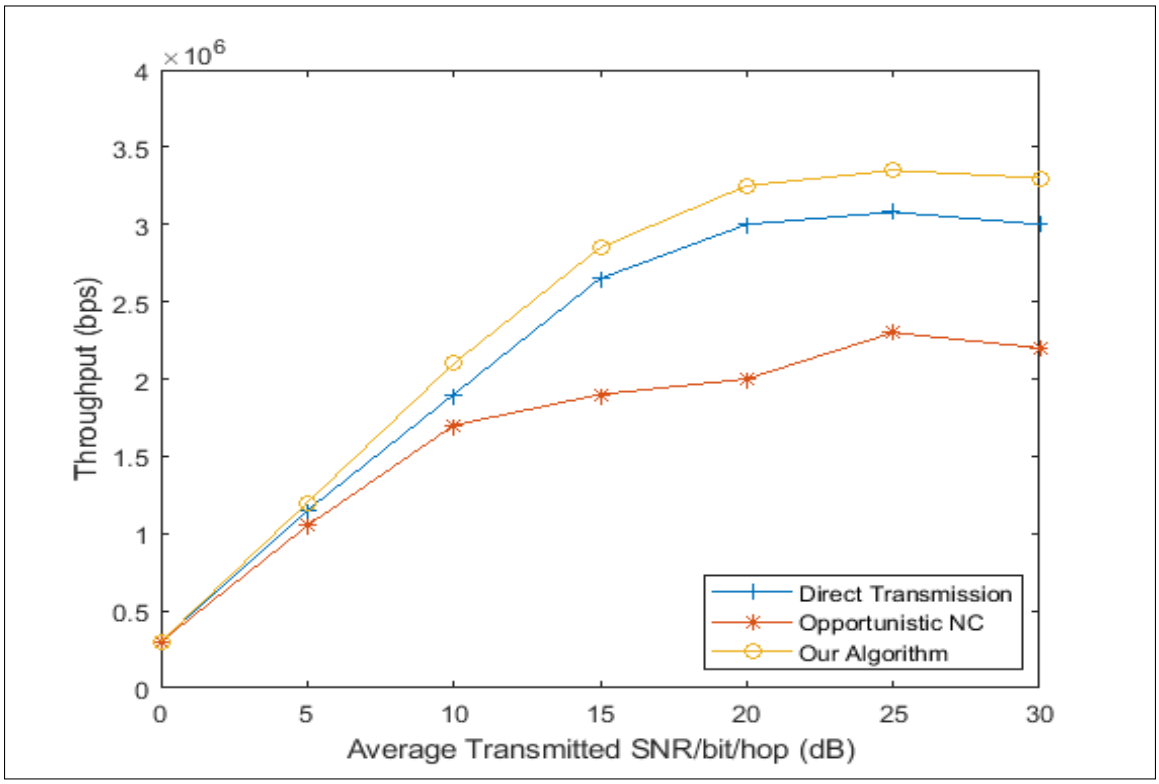

Figure 3.15: System throughput $P_o(E)$=0.1.

Chapter 4: Post-Quantum Hybrid Security Mechanism for MIMO Systems

In the past decade, the world has become gradually connected and the introduction of Internet of Things becomes a widely used notion nowadays in research. While the advancement of technology could put a radio access interface on every device out there and provided reliable communication links, information security took the back seat. Generally, the wireless communication medium security has always been a critical issue since an unprecedented amount of sensitive and private data being transmitted over them. In conventional wireless networks, security issues are primarily handled by the higher-level layer, i.e. application layer, and rely on the computational complexity of an underlying mathematical problem known as cryptographic methods. While they have worked well in practice [40], [41], they might be difficult to implement and may be vulnerable to attacks in some cases since they require a secure channel to exchange keys or certificate management. Most importantly, most public-key cryptosystems are susceptible to large deployment of quantum computers. Current methods rely either on integer factorization, discrete logarithmic, or elliptic curve discrete logarithmic problems which can be solved easily using Shor's algorithm [42].

On the other hand, physical layer security techniques exploit the characteristics of the wireless channel to improve security. It ensures data's security by requiring the latter to be a design constraint rather than a feature. By utilizing physical layer security methods, it becomes more difficult for attackers to decipher transmitted data and more robust to the increase of an adversary computational power. Moreover, physical layer security offers built-in security that is information theoretically unbreakable [43], [44]. Thus, physical layer security is not susceptible to the introduction of quantum computers. The security solutions at the physical layer can complement the cryptographic mechanisms, or work as a standalone solution for a system with strict energy requirements like the ones found in sensor networks. Although promising, physical layer security relies on assumptions about relative quality of channels. When these qualities are partially known or unknown, special handling is required [18]. Furthermore, its perfect secrecy is conditioned on the notion that channels are unknown or noisier at the adversary, which might not be true in all cases [43]. Finally, proving the security guarantee for physical layer is usually a hard task especially for strong secrecy cases [45].

In general, researchers focus on investigating either traditional cryptography or physical layer security and their applications. Nevertheless, there has been little to no effort in investigating a cross-layer security mechanism that combines the advantages of both directions

Table 4.1: Security comparison between MOPRO, Diffie-Hellman + RSA, and the proposed C-MOPRO.

Algorithm	MitM Safe	Info Theoretic	Quantum Resistent	Eve Coverage	Security Loss*
MOPRO	✗	✓	✓	one	50%
DH + RSA	✓	✗	✗	none	0%
C-MOPRO	✓	✓	✓	two	0%

* with the presence of an eavesdropper near Alice or Bob.

and reduce or eliminate the disadvantages of the two schools of security. Therefore, we propose a post-quantum hybrid key agreement with device authentication security mechanism that uses a combination of physical layer security and cryptographic techniques to achieve a powerful security mechanism with reasonable overhead.

Our proposed algorithm (C-MOPRO) is based on the work presented in [46]. However, our work significantly differs from their work in the following aspects: Firstly, our proposed solution assumes an active attacker model while they assume a passive eavesdropper model. The difference is that Eve can do more than just listening to the communication between two legitimate users. Secondly, we implement a digital signature scheme to authenticate the legitimate users to prevent Eve from impersonating any of the original users. This type of attack usually referred to as Man in the Middle (MitM) attack. Specifically, we implement SPHINCS which is a stateless hash-based signature scheme. SPHINCS depends only on the existence of secure hash functions which makes it very adjustable and invulnerable to quantum computing [47]. Finally, we address the issue where one of the legitimate users' keys gets jeopardized resulting in unveiling half of communicated messages to an eavesdropper. The security comparison of our proposed algorithm against other techniques is summarized in Table 4.1.

4.1 Background

4.1.1 Cryptographic Primitives

Considering the wide introduction of quantum computers and its consequences on modern digital signatures, current post-quantum cryptography research proposes SPHINCS as one of the best alternatives. As stated before, SPHINCS is a stateless hash-based signature scheme. In fact, one-time signature (OTS) forms the basic block in all hash-based signatures. Merkle

adopted this scheme to construct a many-time signature scheme [48]. When a Merkle tree is used on top of OTS key pairs, the choice of an OTS key more than once should be avoided. This requires us to store some info, i.e. state, about the keys have already been used making it impractical in some cases. To overcome this problem, Goldreich proposed a scheme that creates a tree in a way that makes the probability of choosing a previously used key significantly small [49]. However, the size of Goldreich's signature is extremely large.

SPHINCS overcomes both challenges; the state and signature size. It does that by combining Goldreich's scheme with Merkle trees and few-time signatures. The authors use Winternitz One-Time Signature (WOTS+)[1] scheme to form the Merkle tree [2]. Also, they propose HORST few-time signature scheme, which is basically a version of HORS [51] with trees, to sign the message digest. Both schemes are defined in Algorithm 2 and Algorithm 3, respectively.

Global parameters: Winternitz parameter $w \in \mathbb{N}$, $w > 1$, message M, security parameter $n \in \mathbb{N}$, input seed $S \in \{0,1\}^n$, $l_1 = \lceil n/log(w) \rceil$, $l = l_1 + \lfloor log(l_1(w-1))/log(w) \rfloor + 1$, $G_\lambda : \{0,1\}^n \to \{0,1\}^{\lambda n}$, $\mathcal{V} : \{0,1\}^n \to \{0,1\}^n$.

Algorithm 2 WOTS+ SIGNATURE

1: **Parameters:** $|M| = n$, bitmasks $\mathbf{r} \in \{0,1\}^{n*(w-1)}$, $c^i(x, \mathbf{r}) = \mathcal{V}(c^{i-1}(x, \mathbf{r}) \oplus r_i)$

2: **Key Generation** $(SK, PK) \leftarrow WOTS.kg(S, \mathbf{r})$: Outputs secret key SK and public key PK

- $SK = (SK_1, .., SK_l) \leftarrow G_l(S)$
- $PK = (PK_1, .., PK_l) = (c^{w-1}(SK_1, \mathbf{r}), .., c^{w-1}\ (SK_l, \mathbf{r}))$

3: **Signing** $\sigma_{WOTS} \leftarrow WOTS.sign(M, S, \mathbf{r})$: Outputs signature σ_{WOTS} for M under SK

- SK and PK are generated on the fly since storage(S) ¡ storage(SK)

4: **Verifying** $PK' \leftarrow WOTS.vf(M, \sigma_{WOTS}, \mathbf{r})$: Outputs PK' that will be compared to PK in SPHINCS algorithm (returns true on equality, and false otherwise)

SPHINCS deploys a hyper-tree of height h that contains d layers of trees of height h/d [14]. In more details, each layer i has $2^{(d-1-i)(h/d)}$ trees. WOTS+ key pairs of the trees on layer $i+1$ are used to sign the roots of layer i trees. The WOTS+ key pair on layer 0 is used to sign a HORST public key. Finally, each HORST key pair is used to sign the message digest.

[1] The authors of [47] slightly deviated from description of WOTS+ in [50].

It is worth noting that a pseudo-randomly generated index is used to choose which trees inside the hyper-tree are used and which HORST key pair is selected. Finally, to verify, a Merkle tree authentication path is provided as part of the signature. SPHINCS is described in Algorithm 4. For more details, readers are referred to [47].

Algorithm 3 HORST SIGNATURE

1: **Parameters:** message length m, $t = 2^{\tau}$ where $\tau \in \mathbb{N}$, $k \in \mathbb{N}$ where $k\tau = m$, bitmasks $\mathbf{Q} \in \{0,1\}^{2n*logt}$, $x \in \mathbb{N} \setminus \{0\}$

2: **Key Generation** $PK \leftarrow HORST.kg(S, \mathbf{Q})$: Outputs public key PK

- $SK = (SK_1, .., SK_t) \leftarrow G_t(S)$
- A tree is constructed using $\mathbf{Q}$ where tree leaves $L_i = \mathcal{V}(SK_i)$ for $i \in [t-1]$
- $PK =$ root node of a binary tree of height $log(t)$

3: **Signing** $(\sigma_{HORST}, PK) \leftarrow HORST.sign(M, S, \mathbf{Q})$: Outputs PK and signature σ_{HORST} for M under SK

- $SK = (SK_1, .., SK_t) \leftarrow G_t(S)$
- $M = (M_0, .., M_{k-1})$ where $|M_i| = log_2(t)$ bits for $i \in [k-1]$
- Determine x such that $k(\tau - x + 1) + 2^x$ is minimal
- $\sigma_{HORST_i} = (SK_{M_i}, Auth_{M_i})$ where $Auth_{M_i}$ is the lower $\tau - x$ elements of the authentication path of leaves $(A_0, .., A_{\tau-1-x})$ for $i \in [k-1]$
- $\sigma_{HORST_k} = 2^x$ nodes of level $\tau - x$ binary tree

4: **Verifying** $PK' \leftarrow HORST.vf(M, \sigma_{HORST}, \mathbf{Q})$: The signature is valid if all nodes and authentication paths agree on the same root PK (i.e. $PK' = PK$)

4.1.2 MIMO Precoding

MIMO precoding is a processing technique which functions as a multi-mode beamformer to support multi-stream data transmission. By allocating appropriate transmission power to data streams, it maximizes the channel throughput. To achieve the optimal MIMO channel capacity, the optimal precoding matrix requires full channel state information at the transmitter (CSIT). Assuming slow frequency non-selective fading, the received signal is described by $\mathbf{y} = \mathbf{Hx} + \mathbf{v}$, where $\mathbf{y}$ is the received signal vector, $\mathbf{H}$ is the MIMO channel matrix, $\mathbf{x}$ is

Algorithm 4 SPHINCS signature

1: **Parameters:** $p = max\{w-1, 2(h+\lceil log(l)\rceil, 2log(t)\}$, $\mathbf{Q} \xleftarrow{\$} \{0,1\}^{pxn}$

2: **Key Generation** $(SK, PK) \leftarrow SPHINCS.kg(1^n)$: Outputs secret key SK and public key PK

- $SK = (SK_1, SK_2, \mathbf{Q})$ where $(SK_1, SK_2) \in \{0,1\}^n$ x $\{0,1\}^n$
- $PK = (PK_1, \mathbf{Q})$ where PK_1 = root node of a binary tree that is built on public keys of WOTS+ key pairs

3: **Signing** $\sigma_{SPHINCS} \leftarrow SPHINCS.sign(M, SK)$: Outputs signature $\sigma_{SPHINCS}$ for M under SK

- $\sigma_{SPHINCS} = (I, \sigma_{HORST}, Auth_i, \sigma_{WOTS_i})$ where I is index, σ_{WOTS_i} is WOTS+ signature per layer i, and $Auth_i$ is the authentication path per layer i

4: **Verifying** $ind \leftarrow SPHINCS.vf(M, \sigma_{SPHINCS}, PK)$: Returns true if the verification algorithm reaches to the same root node in PK_1, otherwise it returns false

the transmitted signal vector, and $\mathbf{v}$ is the white Gaussian noise vector. To obtain the optimal gain, the MIMO channel matrix $\mathbf{H}$ can be decomposed by performing the singular value decomposition (SVD) of the channel matrix as $\mathbf{H} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^H$, where $[.]^H$ is the Hermitian operator, $\mathbf{U}, \mathbf{V}$ are complex unitary matrices and $\mathbf{\Sigma}$ is a matrix whose diagonal elements are the singular values of $\mathbf{H}$. The optimal beam directions with perfect CSIT are matched to the channel right singular vectors $\mathbf{V}$. Therefore, this requires the channel to be approximately constant over a considerably large period as well as a large feedback overhead. Alternatively, WiMAX and LTE systems use a codebook that consists of multiple precoding matrices and their corresponding PMIs, which yields a balance between system performance, equalizer complexity, and the feedback overhead.

The MIMO-OFDM channel matrix $\mathbf{H}$ is estimated at the receiver using the pilot symbols sent by the transmitter. Then, the suboptimal precoding matrix that maximizes the channel capacity is selected by the receiver using the following equation:

$$\max_{\mathbf{F}\in\mathcal{F}} Capacity_{\mathbf{H},\mathbf{F}} = \log_2 \det[\mathbf{I}_n + \frac{E_s}{n_s\sigma^2}\mathbf{F}^H\mathbf{H}^H\mathbf{H}\mathbf{F}] \tag{4.1}$$

where $\mathbf{F}$ is the precoding matrix, $\mathcal{F}$ is the universal codebook, $\mathbf{I}_n$ is the identity matrix and n is the minimum number of antennas at Alice and Bob, E_s is the total energy of the transmitted signal, n_s is the number of data elements, and σ^2 is the noise variance. Finally, the receiver

Table 4.2: C-MOPRO notations.

$\mathbf{G}$	M_AxM_A random unitary complex matrix
$\mathbf{r}$	M_AxN_r complex reference signal
$\mathbf{U}_{B,i}$	M_BxM_B complex unitary matrix
$\mathbf{V}_{B,i}^H$	M_AxM_A complex unitary matrix
$\mathbf{V}_{A,i}$	M_AxM_A complex unitary matrix
$\mathbf{U}_{A,i}^H$	M_BxM_B complex unitary matrix
$\breve{\mathbf{F}}$	M_Bxn_s
$\mathbf{G}_u$	n_sxn_s complex unitary matrix
$\mathbf{s}$	n_sxN_s complex matrix
(SK_B, PK_B)	Bob's secret and public keys
(SK_A, PK_A)	Alice's secret and public keys

sends the corresponding PMI of the suboptimal precoding matrix to the transmitter.

4.2 System Model

The system consists of two legitimate users (Alice and Bob) and an eavesdropper (Eve). The users are connected using wireless MIMO channels $\mathbf{H}_{AB}$, $\mathbf{H}_{AE}$, and $\mathbf{H}_{BE}$. This model is depicted in Fig. 4.1. Alice wants to communicate with Bob confidentially through $\mathbf{H}_{AB}$. Due to the broadcast nature of wireless channels, Eve can listen to the messages originated at Alice and Bob through $\mathbf{H}_{AE}$ and $\mathbf{H}_{BE}$, respectively. It is assumed that the MIMO system uses time division duplexing and the MIMO channel reciprocity holds in the transposed form $\mathbf{H}_{AB}$= $\mathbf{H}_{BA}^T$, where $[.]^T$ is the matrix transpose, along with perfect channel reciprocity. Alice, Bob, and Eve are equipped with M_A, M_B, and M_E number of antennas, respectively.

As in [46], the universal codebook containing precoding matrices and the corresponding precoding matrix indices (PMIs) is accessible to all parties Alice, Bob, and Eve. The channel capacity function used by Alice and Bob is also known to Eve. The mapping between precoding matrix and secret key sequence is a predefined public information. All parties have knowledge of this mapping in advance. Eve is assumed to be an active attacker who will falsify public discussion and/or listen to the communications between Alice and Bob but will not jam the channel.

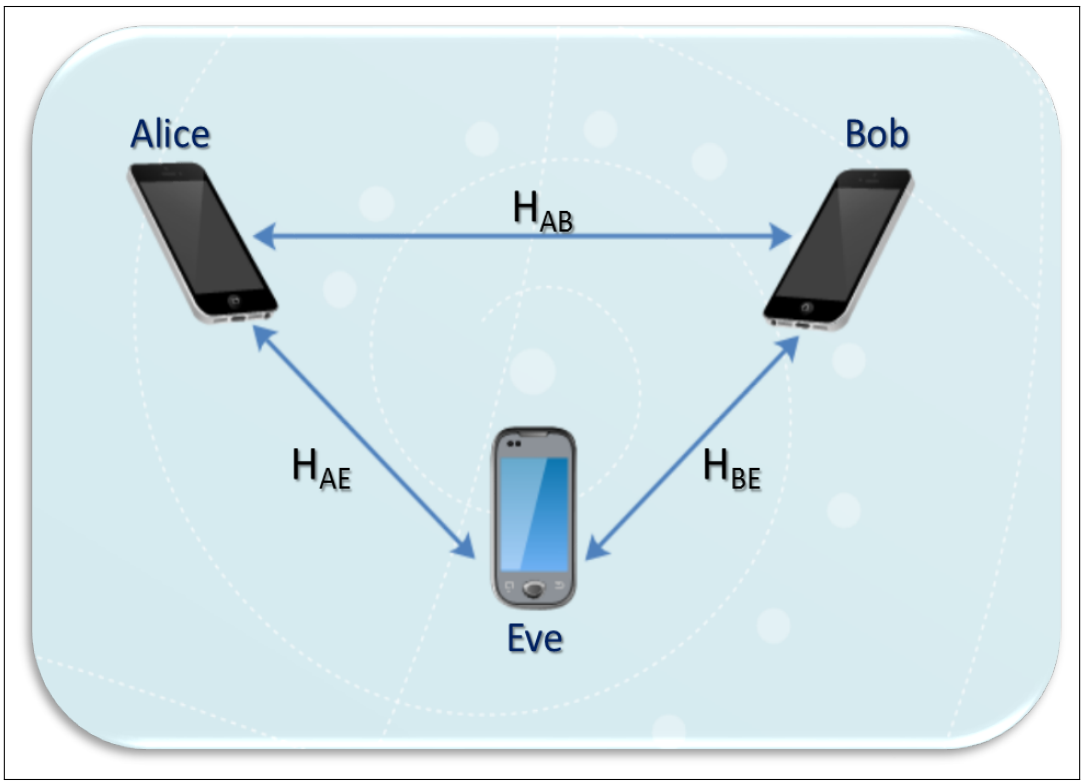

Figure 4.1: System layout.

4.3 Proposed Algorithm

In this section, the proposed algorithm C-MOPRO is detailed. Our proposed solution is based on the MOPRO scheme presented in [22]. The algorithm utilizes complex unitary rotation matrices to hide the secrecy information and exchange secret keys during the communication establishment phase. Although similar, our work differs in the following: 1) It assumes an active attacker model. 2) It addresses the Man in the Middle (MitM) attack. 3) It addresses the issue of exposing half of the secret key. Fig. 4.2 depicts the exchanged messages between the legitimate users and what is heard by Eve. The flow of our algorithm is detailed next and the notation used in the algorithm is defined in Table 4.2.

1. Alice transmits the reference signal $\mathbf{Gr}$ to Bob to estimate the channel. Bob estimates the sub-band i averaged channel $\mathbf{H}_{AB,i}\mathbf{G}_i$ and performs SVD on $\mathbf{H}_{AB,i}\mathbf{G}_i$ to obtain $\mathbf{H}_{AB,i}\mathbf{G}_i = \mathbf{U}_{B,i}\mathbf{\Sigma}_i\mathbf{V}_{B,i}^H\mathbf{G}_i$, where $\mathbf{\Sigma}_i$ is M_BxM_A matrix.

2. Bob generates a secret key $\mathcal{K}_{Bob}$ of c-bits. Bob applies channel coding and obtains the coded sequence $\mathcal{C}_{Bob}$. Based on the codebook used, Bob divides $\mathcal{C}_{Bob}$ into $\lceil \frac{c}{p} \rceil$ groups each denoted $\mathcal{C}_{Bob,i}$.

3. Using $\mathcal{C}_{Bob,i}$ as PMI, Bob finds the corresponding precoding matrix $\mathbf{F}_{B,i}$. Bob appends

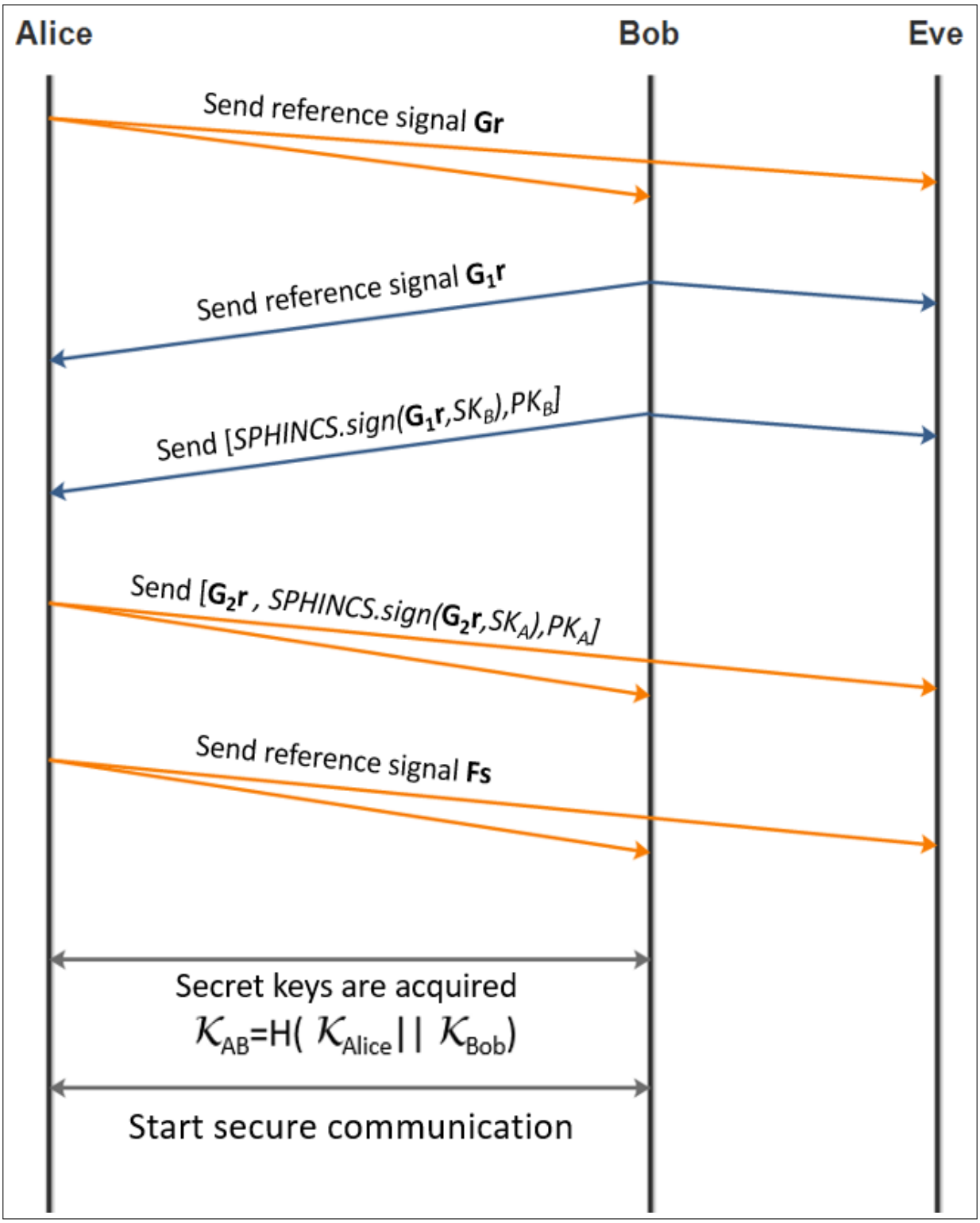

Figure 4.2: C-MOPRO message exchange between Alice and Bob.

Table 4.3: Overhead comparison between MOPRO, Diffie-Hellman + RSA, and the proposed C-MOPRO.

	Alice Overhead		**Bob Overhead**	
Algorithm	**Computation**	**Communication (in bits)**	**Computation**	**Communication (in bits)**
MOPRO	-	$n_b\|\mathbf{Gr}\|$	-	-
DH + RSA	KA: $1.Exp$ SGN: $2.Exp'$+ $2.Hash$	$\|p\| + \|g\| + \|A\|$ + $\|PK_{RSA_A}\|$ + $\|\sigma_{RSA}\|$	KA: $1.Exp$ SGN: $2.Exp'$ +$2.Hash$	$\|B\| + \|PK_{RSA_B}\|$ + $\|\sigma_{RSA}\|$
C-MOPRO	KA: $1.Hash$ SGN: n_bC	$\|PK_{SPH_A}\|$ + $n_b\|\mathbf{Gr}\| + n_b\|\sigma_{SPH}\|$	KA: $1.Hash$ SGN: n_bC	$\|PK_{SPH_B}\| + n_b\|\sigma_{SPH}\|$

KA: Key agreement algorithm. SGN: Digital signature algorithm. Exp: Module exponentiation in DH. Exp': Module exponentiation in RSA. $Hash$: Hash function. p, g, A, B: Parameters for Diffie-Hellman key exchange algorithm, where $|p|$=$|g|$=3072 bits. n_b: Number of sub-bands. PK_{RSA}: RSA public key. PK_{SPH}: SPHINCS public key. σ_{RSA}: RSA signature. σ_{SPH}: SPHINCS signature. C: Cost of SPHINCS-256 signature which consists of 699494 ChaCha12 permutations [14]. For 128-bit post-quantum security: 3072-bit DH, 3072-bit RSA, SPHINCS-256, and SHA-384 are considered [41].

random orthogonal columns to $\mathbf{F}_{B,i}$ to make it a full rank M_BxM_B complex unitary matrix $\hat{\mathbf{F}}_{B,i}$.

4. Bob transmits the rotated reference signal $\mathbf{G}_{1,i}\mathbf{r}$ to Alice, where $\mathbf{G}_{1,i} = \mathbf{U}^*_{B,i}\hat{\mathbf{F}}^H_{B,i}$ and $[.]^*$ is the matrix conjugation. Then, Alice estimates PMI of the ith sub-band from $\mathbf{H}_{BA,i}\mathbf{G}_{1,i}$.

5. Bob generates $(SK_B, PK_B) \leftarrow SPHINCS.kg(1^n)$. Bob transmits $[SPHNCS.sign(\mathbf{G}_{1,i}\mathbf{r}, SK_B)$ $, PK_B]$ to Alice. Then, Alice verifies Bob on the ith sub-band using $SPHINCS.vf(\mathbf{G}_{1,i}\mathbf{r}$, $\sigma_{SPHINCS}, PK_B)$.

6. Steps 3-5 are repeated for all sub-bands. Alice combines all the collected PMIs to form $\mathcal{C}_{Bob}$ and then obtain $\mathcal{K}_{Bob}$. Alice generates a secret key $\mathcal{K}_{Alice}$ of c-bits.

7. Alice applies channel coding and obtains the coded sequence $\mathcal{C}_{Alice}$ and divides $\mathcal{C}_{Alice}$ into $\lceil\frac{c}{p}\rceil$ groups each denoted $\mathcal{C}_{Alice,i}$. Using $\mathcal{C}_{Alice,i}$ as PMI, Alice finds the corresponding precoding matrix $\mathbf{F}_{A,i}$. Bob appends random orthogonal columns to $\mathbf{F}_{A,i}$ to make it a full rank M_AxM_A complex unitary matrix $\hat{\mathbf{F}}_{A,i}$.

8. Alice performs SVD on $\mathbf{H}_{BA,i}\mathbf{G}_{1,i}$ to obtain $\mathbf{H}_{AB,i}^{T}\mathbf{G}_{1,i} = \mathbf{V}_{A,i}^{*}\mathbf{\Sigma}_{i}^{T}\mathbf{U}_{A,i}^{T}\mathbf{G}_{1,i}$, where $\mathbf{\Sigma}_i$ is M_AxM_B diagonal matrix. Alice transmits the rotated reference signal $\mathbf{G}_{2,i}\mathbf{r}$ to Bob, where $\mathbf{G}_{2,i} = \mathbf{V}_{A,i}\hat{\mathbf{F}}_{A,i}^{H}$. Bob estimates PMI of the ith sub-band from $\mathbf{H}_{AB,i}\mathbf{G}_{2,i}$.

9. Alice generates $(SK_A, PK_A) \leftarrow SPHINCS.kg(1^n)$ and sends $[SPHNCS.sign(\mathbf{G}_{2,i}\mathbf{r}, SK_A)$ $, PK_A]$. Then, Bob verifies Alice on the ith sub-band using $SPHINCS.vf(\mathbf{G}_{2,i}\mathbf{r}, \sigma_{SPHINCS}$ $, PK_A)$.

10. The steps are repeated for all sub-bands. Alice combines all the collected PMIs to form $\mathcal{C}_{Alice}$ and then obtain $\mathcal{K}_{Alice}$.

11. Alice and Bob apply a cryptographic hash function on the concatenation of Alice and Bob keys. A shared secure key is defined by $\mathcal{K}_{AB} = H(\mathcal{K}_{Alice}||\mathcal{K}_{Bob})$.

12. Alice finds the optimal precoding matrix to achieve MIMO channel capacity using: $\breve{\mathbf{F}} = \max_{\mathbf{F}\in\mathcal{F}} Capacity_{\mathbf{H},\mathbf{F}} = \log_2 \det[\mathbf{I}_n + \frac{E_s}{n_s\sigma^2}\mathbf{F}^H\mathbf{H}^H\mathbf{H}\mathbf{F}]$. Alice generates $\mathbf{G}_u$ and creates $\mathbf{F} = \breve{\mathbf{F}}\mathbf{G}_u$. Alice transmits the reference signal $\mathbf{F}\mathbf{s}$ and Bob estimates the channel $\mathbf{H}_{AB}\mathbf{F}$.

4.4 Security Analysis

In this section, we discuss the security guarantee of the proposed solution. The security guarantee of C-MOPRO, DH + RSA, and MOPRO is summarized in Table 4.1. By deploying the physical layer security mechanism, exchanging uniformly distributed secrets keys is made possible during the channel establishment phase. Furthermore, the use of the unitary rotation matrices prevents Eve from acquiring either $\mathbf{H}_{AE}$ or $\mathbf{H}_{BE}$ since only the rotated channel is used to exchange messages. This renders Eve attempts to reconstruct the complete channel between Alice and Bob useless and provides additional security to the communication channel. However, based on Eve's location there might be a risk of exposing half of the secret key bits. If Eve places itself near either Alice or Bob, then the channel experienced by Eve will be close to either one of the legitimate users. For example, if Eve placed itself close to Bob then $\mathbf{H}_{AE}\mathbf{G}_2 \simeq \mathbf{H}_{AB}\mathbf{G}_2$ and by performing PMI estimation Eve can obtain $\mathcal{K}_{Alice}$.

For physical layer security mechanism to be information theoretically unbreakable, it has to satisfy the strong secrecy condition defined as $\lim_{n\to\infty} I(W|Z^n) = 0$. This requires that the mutual information between each bit of the message W and the observed n-length cipher Z^n at Eve to be zero, i.e. no information leakage about the message when the transmitted cipher is observed by Eve [52]. To remedy this, we propose that both legitimate users should apply a universal hash function on the concatenation of both Alice and Bob keys to generate a shared

key $\mathcal{K}_{AB} = H(\mathcal{K}_{Alice}||\mathcal{K}_{Bob})$. Thus, if Eve was successful in obtaining one of the legitimate users key, Eve will not be able to obtain the shared key. This is due to the fact that any small change in the hash function input will cause the output to change drastically. Nevertheless, the security of C-MOPRO can be compromised if two active attackers placed themselves near Alice and Bob simultaneously. Still, this requires the two attackers to exchange data risking alerting either Alice or Bob which might result in terminating the communication.

In addition, implementing the physical layer security mechanism allows us to authenticate the legitimate users during channel establishment phase. The wireless channel between the legitimate users becomes decodable after the transmission of the rotated reference signals and hence we can authenticate transmitted signals to prevent MitM attack. Alternatively, traditional cryptography usually authenticates and secures the channel after the channel has been established and usually does not concern itself with this process. With the rise in fear of the inevitable large-scale implementation of quantum computers, many of the digital signature schemes that rely on the integer factorization problem, the discrete logarithm problem, or the elliptic curve discrete logarithm problem can be solved easily. Therefore, we opted to implement SPHINCS to authenticate the legitimate users. The authors of SPHINCS proved its security against quantum attacks since it only depends on the usage of secure cryptographic hash functions.

Finally, it is important to note that MOPRO and C-MOPRO provide information theoretic security. The authors in [46], showed that using the rotation matrices decreases Eve's knowledge about the channel.

$$\bar{H}(\mathbf{h}_{AB}|\mathbf{h}_{AE}) \leq \bar{H}(\mathbf{h}_{AB}\mathbf{G}_2|\mathbf{h}_{AE}\mathbf{G}_2) \tag{4.2}$$

and

$$\bar{H}(\mathbf{h}_{BA}|\mathbf{h}_{BE}) \leq \bar{H}(\mathbf{h}_{BA}\mathbf{G}_1|\mathbf{h}_{BE}\mathbf{G}_1) \tag{4.3}$$

where $\bar{H}$ is the entropy and $\mathbf{h}$ is the simplified channel matrix.

4.5 Performance Analysis

Overhead comparison between MOPRO, Diffie-Hellman + RSA, and the proposed C-MOPRO is detailed in Table 4.3. The table shows the computation and communication overhead for Alice and Bob, respectively. In MOPRO, the generation of Alice and Bob respective keys requires no computation overhead in terms of the number of exponentiations and hash operations. Since their secret keys are embedded into the required reference signals to estimate the channel, one of the users does not acquire communication overhead. However, the other user

will need to send additional $n_b\mathbf{Gr}$ reference signals to communicate its secret key securely.

Alternatively, the Diffie-Hellman + RSA algorithm requires each Alice and Bob one exponentiation to agree on a key. Additionally, it requires each Alice and Bob one exponentiation and one hash function operation to authenticate or verify the exchanged messages. Furthermore, the communication overhead associated with Diffie-Hellman + RSA algorithm is the result from communicating Diffie-Hellman parameters, RSA public keys, and RSA signature.

On the other side, our proposed C-MOPRO algorithm requires each Alice and Bob one hash function operation to agree on a shared secret key. Also, it requires each Alice and Bob n_bC to authenticate and verify the exchanged signals. As in MOPRO, our proposed solution requires additional $n_b\mathbf{Gr}$ reference signals to transmit the second secret key. On top of that, C-MOPRO needs to communicate Alice/Bob public keys and signatures to authenticate the messages. It is important to highlight that the parameter n_b in MOPRO and C-MOPRO is a design choice and depends on the total bandwidth, the sub-band bandwidth, and the desired length of the secret key. In fact, selecting an appropriate number of sub-bands is critical since it affects the computation and communication overhead. Hence, in our future work, we aim to find the optimal n_b that results in a reasonable overall overhead and yet maintains high system capacity.

The main contributing factor in C-MOPRO overhead is due to SPHINCS which is computationally costly when compared to traditional digital signatures. Nevertheless, in the age of quantum computing, SPHINCS and other post-quantum schemes must be used instead of traditional cryptography signatures, e.g. RSA. As a matter of fact, all post-quantum hash-based signatures result in higher overhead compared to traditional signatures [47], [53]. This is the tradeoff between security and performance. Other than SPHINCS overhead, C-MOPRO has a reasonable computational and communicational overhead when compared to post-quantum key exchange algorithms. This is since the key agreement in C-MOPRO is done during the channel establishment phase and it does not require a generation of a public and private key pair to agree on a secret key. In addition, it has been established that many post-quantum key exchange protocols are computationally costly [54], [55]. For example, Supersingular Isogeny Diffie-Hellman (SIDH) key exchange which serves as a replacement to DH takes 303ms to agree on a key[2] [56]. This does not include the time needed for channel establishment and message authentication.

[2]This was measured on Macbook Pro Intel Core i5-2415M @ 2.4 GHz.

4.6 Conclusion

In this paper, we proposed the C-MOPRO algorithm which is a post-quantum hybrid security algorithm. This cross-layer security mechanism combines cryptographic techniques and physical layer security to achieve a powerful security mechanism with a reasonable overall overhead. In this scheme, the key agreement is accomplished during the channel establishment phase. Also, during this phase, we address MitM attack using SPHINCS digital signature. Furthermore, we tackle the problem where half of the secret key bits gets compromised when Eve is located near either Alice or Bob. This is done using a universal secure hash function that guarantees the security of the shared secret key even if half of the secret key bits is exposed.

Chapter 5: Conclusion and Future work

5.1 Conclusion

In this dissertation, we explored multiple coding techniques to reduce energy consumption, improve performance, and secure wireless sensor networks specifically and ad-hoc networks in general. With the introduction of Internet of Things (IoT) and 5G technologies, wireless sensor networks are quickly emerging as an important and key technology in the future. From their ability to sense, process, and communicate data among them to being low-powered, self organizing, and cost effective. Their characteristics made them a great tool for many applications, they already have a role in connecting homes, cars, surveillance systems, early earthquake and forest fire detection. However, due to their limited power and processing energy, they suffer to maintain acceptable performance and connectivity especially when deployed in harsh environment. In this research, we demonstrated novel techniques that can help improve their performance while reducing energy consumption. The contribution of this work is summarized below.

- We propose a novel approach to error correction codes in wireless sensor network. We introduce a modification to Reed-Solomon decoding algorithm which allows errors to occur in data without sacrificing the total integrity of the data. We show that by deploying such mechanisms, we can reduce the total energy required to deliver data at their destination by reducing the decoding energy per symbol/bit. However, we conclude that the savings is minimal taken into account the degradation occurs in performance. As well, the requirement for having a non-binary error correction codes to achieve and replicate savings in decoders. As of now, the only well established non-binary error correction codes are Reed-Solomon codes.

- Opportunistic network coding was introduced as mean to increase network throughput by XORing two or more packets together. However, it has been shown that ONC's bit error rate (BER) and throughput suffer when implemented in wireless network with noise model. We propose a modification on opportunistic network coding (ONC) using diversity coding and cooperation, as well as, limiting the number of packets that can be network-coded together to three and only encode packets that were received by relay nodes directly. We show that using such techniques we can alleviate the issues that plague ONC when implemented in noisy networks. later on, we study the effect of

link outages/mobility on proposed solution and show that our proposed solution can accommodate up to one link failure.

- We study the security of ad-hoc networks and propose a post-quantum hybrid security mechanism. We propose a security mechanism that take advantage of the wireless medium hereditary nature and cryptography techniques. This state of art protocol is able to overcome the presence of adversary eavesdropper and address man in the middle attack. Our security mechanism uses a combination of physical layer and cryptographic security techniques to provide best effort security.

5.2 Future Work

In this section, we will summarize the future work related to the research topics of this dissertation. We will discuss future enhancements and possible research tracks that are of interest in their designated sections.

1. Modified Reed-Solomon Decoding algorithm:

   (a) In chapter 2, we profiled the power consumption of RS decoding algorithm and summarized the power consumption of each of its component in Table 2.4. We can see that 85% of power is consumed in identifying if errors took place and identifying errors locations in the codewords. Therefore, developing methods and algorithms that reduce the power consumption related to calculating the syndrome and locating the errors will help in increasing the overall energy savings in partial error correction algorithms.

   (b) A possible research direction and a good alternative to RS codes are low-density parity check (LDPC) codes. These codes have gained a great momentum and became essential part for many applications in wireless communication. On the contrary, non-binary LDPC codes are uncharted territory. In [57, 58], non-binary LDPC codes shown to outperform its binary variation as the finite field increases, have better convergence, and requires less iterations to decode codewords. This becomes on the expense of higher complexity. In [59], a fast Fourier transforms method is proposed which reduces the decoder complexity, allowing non-binary codes to be a good alternative.

   (c) A common error correction code used extensively in wireless communication systems such as 3G and 4G is Turbo codes. Non-binary turbo codes have shown to have several advantages [60]. Nevertheless, non-binary turbo received little attention in research due to complexity [60–62].

2. Diversity Opportunistic Network Coding with Cooperation:

   (a) As we discussed in Chapter 3, we limited the number of network coded packets to three. Even though, we showed that the probability to encode more than three packets is insignificant it is still important to address the network performance without such limitation. Additionally, we are interested in evaluating the performance after reducing the restrictions proposed in this research. It is important to examine the impact of removing these restrictions on the total performance of the network. Additionally, we did not take into account the additional processing needs of our proposed solution and how that impact the power consumption of the network.

   (b) We built our simulation on top using a simulator based on COPE which utilizes the IEEE802.11. In future work we want to implement this work with more appropriate system model such as Zigbee which utilize a similar framework but is more focused on providing connectivity to ad-hoc and wireless sensor networks.

3. Post-Quantum Hybrid Security Mechanism for MIMO Systems:

   (a) As we discussed in Chapter 4, post-quantum signatures such as SPHINCS are computationally and communicationally expensive [63, 64]. Hence, other post-quantum signatures can be explored and compared to SPHINCS in term of computational and communicational overhead and security. Another research direction can be investigated is to implement C-MOPRO using SPHINCS$^+$ framework [65]. In [65], the author proposed a more optimized version of SPHINCS in term of speed, signature sized, and security which makes SPHINCS$^+$ a good candidate for improving C-MOPRO performance and overhead.

Bibliography

[1] I. F. Akyildiz, W. Su, Y. Sankarasubrarmaniarn, and E. Cayirei. Wireless sensor networks: A survey. *Computer Networks (Elsevier) Journal*, pages 393–422, March 2002.

[2] M. Murshed and A. Allen. Energy efficient dynamic routing protocol for wireless sensor networks. *Conference on Computer Science and Information Technology (CCSIT)*, LNISCT 84:41–52, 2012.

[3] M. A. M. Vieira, Claudionor N. C. Jr., D. C. da S. Jr., and J. M. da Mata. Survey on wireless sensor network devices. *IEEE Conferences on Emerging Technologies and Factory Automation*, 1:537–544, Nov 2003.

[4] S. Lanm, M. Qilong, and J. Du. Architecture of wireless sensor networks for environmental monitoring. *IEEE International Workshop on Education Technology and Training and International Workshop on Geoscience and Remote Sensing*, 1:579–582, Dec 2008.

[5] G. Hoblos, M. Staroswiecki, and A. Aitouche. Optimal design of fault tolerant sensor networks. *IEEE International Conference on Control Application*, Sept 2000.

[6] N. Bulusu, D. Estrin, L. Girod, and J. Heidemann. Scalable coordination for wireless sensor networking: self-configuring localization systems. *ISCTA 2001, Ambleside, UK*, July 2001.

[7] N. M. Freris, H. Kowshik, and P. R. Kumar. Fundamentals of large sensor networks: connectivity, capacity, clocks, and computation. *Proceeding of the IEEE*, 98(11):1828–1846, November 2010.

[8] M. Xiang, L. Sun, and L. Li. Survey on the connectivity and coverage in wireless sensor networks. *2011 7th international conference on wireless communications and networking and mobile computing (WiCOM)*, September 2011.

[9] P. Gupta and P. R. Kumr. Critical power for asymptotic connectivity in wireless networks. *in stochastic analysis, control optimization, and applications: A volume ub honor of W. H. Fleming, W. M. McEneany, G. Yin, and Q. Zhang, Eds. Boston*, pages 547–566, September 1998.

[10] M. Penrose, editor. *Random Geometric Graph*. Oxford University Press, 2003.

[11] F. Xue and P. R. Kumar. The number of neighbors needed for connectivity of wireless network. *Wireless Network*, 10(2):169–181, March 2004.

[12] P. Gupta and P. R. Kumar. The capacity of wireless networks. *IEEE Trans Information Theory*, 46(2):288–404, March 2000.

[13] A. Agarwal and P R. Kumar. Improved capacity bounds for wireless networks. *Wireless communication mobile computation*, 4:251–261, 2004.

[14] Y. Qassim and M.E. Magana. Error-tolerant non-binary error correction code for low power wireless sensor networks. *International Conference on Information Networking (ICOIN)*, pages 23–27, February 2014.

[15] N. Sadeghi, K. Iniewski, S. Howard, V. Gaudet, S. Kasnavi, and C. Schlegel. Analysis of error control code use in ultra-low-power wireless sensor networks. *IEEE ISCAS*, September 2006.

[16] Z. H. Kashani and M. Shiva. Channel coding in multi-hop wireless sensor networks. *In Proceedings of* 6^{th} *International Conference on ITS communications*, June 2006.

[17] B. Shen and A. Abedi. Error correction in heterogeneous wireless sensor networks. *Biennial Symposium on Communications*, June 2008.

[18] G. Balakrishan, M. Yang, Y. Jiang, and Y. Kim. Performance analysis of error control codes for wireless sensor networks. *IEEE International conf. on inf. Tech.*, April 2007.

[19] O. Bredtmann and A. Czylwik. Truncated convolutional codes as a new approach of unequal error protection. *Vehicular Technology Conference Fall (VTC 2010-Fall), 2010 IEEE 72nd*, pages 1–5, Sept 2010.

[20] Zhiyuan Yan and B.W. Suter. Unequal error protection for noncoherent random linear network coding. *Information Sciences and Systems (CISS), 2011 45th Annual Conference on*, pages 1–6, March 2011.

[21] A.E. Babiker and M.N.B. Zakaria. An efficient energy two mode error correction technique in underwater wireless sensor networks. *Information Technology (ITSim), 2010 International Symposium in*, 2:580–585, June 2010.

[22] T. Rappaport, editor. *Wireless Communication: Principles and Practice*. Prentice Hall, 1996.

[23] C.K.P. Clarke. Reed-solomon error correction. *BBC R&D White paper, WHP 031*, 2002.

[24] S Lin and D. Costello, editors. *Error Control Coding*. Prentice Hall, 2004.

[25] E. Fujiwara, editor. *Code Design for Dependable Systems*. Wiley-Interscience, 2006.

[26] R. Ahlswede, N. Cai, S. Li, and R. W. Yeung. Network information flow. *IEEE Trans. Inf. Theory*, 46(4):1204–1216, 2000.

[27] A. Douik, S. Sorour, T.Y. Al-Naffouris, and M. Alouini. Instantly decodable network coding: From centralized to device-to-device communications. *IEEE Communications Surveys Tutorials*, 19(2):1201–1224, 2017.

[28] E. Ayanoglu, Chih-Lin I, R. D. Gitlin, and J. E. Mazo. Diversity coding: using error control for self-healing in communication networks. *Proceedings. IEEE INFOCOM '90: Ninth Annual Joint Conference of the IEEE Computer and Communications Societies*, 1:95–104, June 1990.

[29] E. Ayanoglu, Chih-Lin I, R. D. Gitlin, and J. E. Mazo. Diversity coding for transparent self-healing and fault-tolerant communication networks. *IEEE Transactions on Communications*, 41(11):1677–1686, Nov 1993.

[30] S. N. Avci and E. Ayanoglu. Optimal algorithms for near-hitless network restoration via diversity coding. *IEEE Transactions on Communications*, 61(9):3878–3893, Sep. 2013.

[31] G. E. Arrobo and R. D. Gitlin. Minimizing energy consumption for cooperative network and diversity coded sensor networks. *2014 Wireless Telecommunications Symposium*, pages 1–7, April 2014.

[32] N. I. Sulieman, E. Balevi, K. Davaslioglu, and R. D. Gitlin. Diversity and network coded 5g fronthaul wireless networks for ultra reliable and low latency communications. *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, pages 1–6, Oct 2017.

[33] S. Katti, H. Rahul, D. Katabi, W. Hu, M. Medard, and J. Crowcroft. Xors in the air: Practical wireless network coding. *ACM SIGCOMM*, 36(4):243–254, 2006.

[34] S. Katti, D. Katabi, W. Hu, H. Rahul, and M. Medard. The importance of being opportunistic: Practical network coding for wireless environments. *Johns Hopkins University, Department of Computer Science*, 2005.

[35] A. Sendonaris, E. Erkip, and B. Aazhang. User cooperation diversitypart i: System description. *IEEE Transaction on Communication*, 51:1927, November 2003.

[36] A. Sendonaris, E. Erkip, and B. Aazhang. User cooperation diversitypart ii: Implementation aspects and performance analysis. *IEEE Transaction on Communication*, 51:1939, November 2003.

[37] J. N. Laneman, D. N. C. Tse, and G. W. Wornell. Cooperative diversity in wireless networks: Efficient protocols and outage behavior. *IEEE Trans. on Inf. Theory*, 50(12):3062–3080, December 2004.

[38] Z. Li and B. Li. Network coding the case for multiple unicast sessions. *Proc. 42nd Allerton Conf. Commun., Control, and Comput*, 2004.

[39] P. Poocharoen, M. E. Magaa, and E. X. Alban. Partial network coding with cooperation: A cross-layer design for multi-hop wireless networks. *International Conference on Ultra Modern Telecommunications*, October 2009.

[40] R. Rivest, A. Shamir, and L. Adleman. A method for obtaining signatures and public-key cryptosystems. *Commun. ACM*, 21(2):120–126, Feb 1978.

[41] W. Diffie and M. Hellman. New directions in cryptography. *IEEE Trans. Inf. Theory*, 22(6):644–654, Nov. 1976.

[42] P. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. Computing 26th*, pages 1484–1509, 1997.

[43] A. Wyner. The wire-tap channel. *Bell Syst. Tech. J.*, 54:1355–1387, 1975.

[44] S. Cheong and M. Hellman. The gaussian wire-tap channel. *IEEE Trans. Inf. Theory*, IT-24(4):. 451–456, Jul. 1978.

[45] W. Trappe. The challenges facing physical layer security. *IEEE Communications Magazines*, 53(6):16–20, June 2015.

[46] C. Wu, P. Lan, P. Yeh, C. Lee, and C. Cheng. Practical physical layer security schemes for mimo-ofdm systems using precoding matrix indices. *IEEE Journal on Selected Areas in Communications*, 31(9):1687–1700, Sept. 2013.

[47] D. Bernstein, D. Hopwood, A. Hulsing, T. Lange, R. Niederhagen, L. Papachristodoulou, M. Schneider, P. Schwabe, and Z. Wilcox-O'Hearn. Sphincs: Practical stateless hash-based signatures. *EUROCRYPT*, 9056(8):368–397, April 2015.

[48] R. Merkle. A certified digital signature. *Advances in Cryptology - CRYPTO '89*, 435:218–238, 1990.

[49] O. Goldreich. Two remarks concerning the goldwasser-micali-rivest signature scheme. *Advances in Cryptology - CRYPTO '86*, 263:104–110, 1990.

[50] A. Hulsing. W-OTS+ shorter signatures for hash-based signature schemes. *Africacrypt*, 7918:173–188, 2013.

[51] L. Reyzin and N. Reyzin. Better than biba: Short one-time signatures with fast signing and verifying. *Information Security and Privacy*, 2384:1–47, 2002.

[52] U. Maurer and S. Wolf. Information-theoretic key agreement: From weak to strong secrecy for free. *EUROCRYPT*, 1807:351–368, 2000.

[53] T. Eisenbarth, I. Maurich, and X. Ye. Faster hash-based signatures with bounded leakage. *Selected Areas in Cryptography 20th International Conference*, pages 223–243, August 2013.

[54] D. Jao and L. Feo. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. *Post-Quantum Cryptography 4th International Workshop*, pages 19–34, 2011.

[55] C. Delfs and S. Galbraith. Computing isogenies between supersingular elliptic curves over $\mathbb{F}_p$. *Codes and Cryptography Designs*, 78(2):425–440, 2014.

[56] R. Azarderakhsh, D. Fishbein, and D. Jao. Efficient implementations of a quantum-resistant key-exchange protocol on embedded systems. *Citeseer*.

[57] M.C. Davey and D.J. Mackay. Low density parity check codes over gf(q). *IEEE Information Theory Workshop*, page 70, 1998.

[58] R. A. Carrasco and M. Johnston, editors. *Non-Binary Error Control Coding for Wireless Communication and Data Storage*. Wiley, 2008.

[59] L. Barnault and D. Declerq. Fast decoding algorithm for ldpc over gf(2q). *IEEE Information Theory Workshop*, pages 70–72, 2003.

[60] C. Berrou, M. Jezequel, C. Douillard, and S. Kerouedan. The advantages of non-binary turbo codes. *IEEE Information Theory Workshop*, 2001.

[61] C. Berrou and M. Jezequel. Non-binary convolutional codes for turbo coding. *IET Electronics Letters*, 35(1):3940, 1999.

[62] J. Berkmann. On turbo decoding of non-binary turbo codes. *IEEE Communications Letters*, 2(4):9499, 1999.

[63] A. Yener and S. Ulukus. Wireless physical-layer security: lessons learned from information theory. *Proceedings of the IEEE*, 103(10):1814–1825, Oct. 2015.

[64] U.S. National Security Agency. Commercial national security algorithm suite and quantum computing FAQ, Jan 2016.

[65] D. Bernstein, A. Hulsing, and S. Kolbl. The sphincs+ framework. *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, pages 2129–2146, November 2019.