Hanze
University of Applied Sciences
Groningen

Overall Power Optimization of Thread Mesh Wireless Networks

by

Md Mazedul Islam Khan

GRADUATION REPORT

Submitted to
Hanze University of Applied Sciences Groningen

in partial fulfillment of the requirements
for the degree of

Master Smart Systems Engineering

Groningen
2023

ABSTRACT

This research investigates power optimization in Thread mesh wireless networks through an algorithmic approach, aiming to reduce overall power consumption while maintaining reliable network performance. Transmission power serves as a key parameter for achieving energy efficiency, and the study focuses on two algorithmic approaches: the Monte Carlo Method (MCM) and the Genetic Algorithm (GA). The research involves determining the optimal network configuration and transmission power constraints, selecting appropriate hardware, building the network, and developing the algorithms. Data is collected and analyzed from various network modes and devices across two locations, including lab and home environments, to ensure diverse and representative results. MCM emphasizes optimal network configuration alongside initial transmission power, while GA targets optimal transmission power settings. The findings indicate that both MCM and GA outperform the maximum method in power optimization, with GA offering the best results. By effectively minimizing energy usage, GA ensures network performance is not compromised. The research emphasizes the importance of sustainability by promoting energy-efficient solutions that minimize environmental impact. The project's focus on energy efficiency and reduced power consumption makes it environmentally friendly and sustainable, contributing to reduced energy waste and lowering the carbon footprint associated with Internet of Things (IoT) networks. Additionally, the research process involves the application and development of professional skills, such as data analysis, algorithm design, and critical thinking, to ensure the reliability and relevance of the results. While the ethical aspects of the research may not be directly evident, the focus on sustainability and responsible technological development inherently involves ethical considerations, such as resource conservation and minimizing negative impacts on society and the environment. The findings contribute to the development of energy-efficient IoT networks and serve as a foundation for further exploration into power optimization techniques, encouraging the expansion of sustainable IoT ecosystems.

Keywords: Thread mesh network, parameter optimization, power optimization, transmission power, monte carlo method optimization, genetic algorithm optimization, MOOD-Sense.

DECLARATION

I hereby certify that this report constitutes my own product, that where the language of others is set forth, quotation marks so indicate, and that appropriate credit is given where I have used the language, ideas, expressions or writings of another.

I declare that the report describes original work that has not previously been presented for the award of any other degree of any institution.

Signed,

Md Mazedul Islam Khan

ACKNOWLEDGEMENTS

I want to express my profound gratitude to my research supervisor, Dr. Mart Johan Deuzeman, Dr. Bryan Williams, and Dr. Jair Adriano Lima Silva, for their invaluable guidance, support, and mentorship throughout this research project. Their expertise, insights, and encouragement have been instrumental in shaping my work and helping me achieve my objectives.

My sincere appreciation goes to my colleagues and fellow researchers who have provided constructive feedback, shared their knowledge, and offered technical assistance when needed. Their collaboration has enriched my understanding of the subject matter and contributed to the quality of my research.

I am also grateful to my friends and family for their unwavering support and understanding during this journey. Their belief in me and constant encouragement have been essential in helping me maintain focus and motivation throughout the process.

Lastly, I would like to thank the staff and faculty at Hanze University of Applied Sciences, Groningen, for creating a conducive environment for learning and research. Their dedication to providing resources and fostering a collaborative atmosphere has played a significant role in the success of this project.

Thank you all for your invaluable contributions to my research.

Contents

List of Tables

List of Figures

Chapter 1

Rationale

1.1 Introduction

The research project, titled “Overall Power Optimization of Thread Mesh Wireless Networks,” is a child project within the broader MOOD-Sense initiative. The MOOD-Sense project employs IoT devices to detect and predict challenging behavior in dementia patients. The project aims to develop an early warning system combining sensors, artificial intelligence, and wireless communication to provide feedback for healthcare professionals and improve patient care and safety [1]. To further enhance the connectivity and scalability among IoT devices in the MOOD-Sense project, a new network protocol called Thread is proposed to be implemented. Thread is a low-power, IPv6-based, mesh networking protocol specifically designed for IoT applications, offering secure, reliable, and efficient communication. It supports self-healing networks with robust routing capabilities and features like end-to-end encryption, making it an ideal choice for the MOOD-Sense initiative [2].

The primary focus of this child project is to optimize the energy efficiency of the wireless Thread network protocol utilized by various wireless sensors and MOOD-Sense projects. To achieve this goal, the project examines transmission power network parameters and configuration aspects, such as device types, path loss, positions, and Received Signal Strength Indicator (RSSI). Thread supports multiple device types, including border routers, leader, routers, and end devices, each serving distinct roles in the network. Border routers enable communication between the Thread network and external networks, the leader is responsible for managing network-wide configurations, routers facilitate data routing within the network, and end devices are typically low-power devices that transmit and receive data [3]. By optimizing the configuration of these devices, the project aims to enhance energy efficiency. Through the establishment of a Thread network and the employment of an algorithmic approach using appropriate hardware, this research investigates the impact of transmission power network parameter optimization on maintaining reliable communication between devices while minimizing power consumption. Ultimately, the project aims to develop more energy-efficient IoT networks to improve the performance of the MOOD-Sense initiative.

1.2 Present Situation

The MOOD-Sense research project originally planned to use three wireless communication technologies: Bluetooth Low Energy (BLE), ZigBee, and Wi-Fi for network communication. However, without a central network protocol, various subprojects within MOOD-Sense, such as dementia patient behavior registration and environmental context monitoring, are being carried out separately. This separation leads to disconnected devices and makes data sharing and integration difficult. The current situation in the lab setup can be visualized as a diagram showing isolated subprojects and devices without an integrated network.

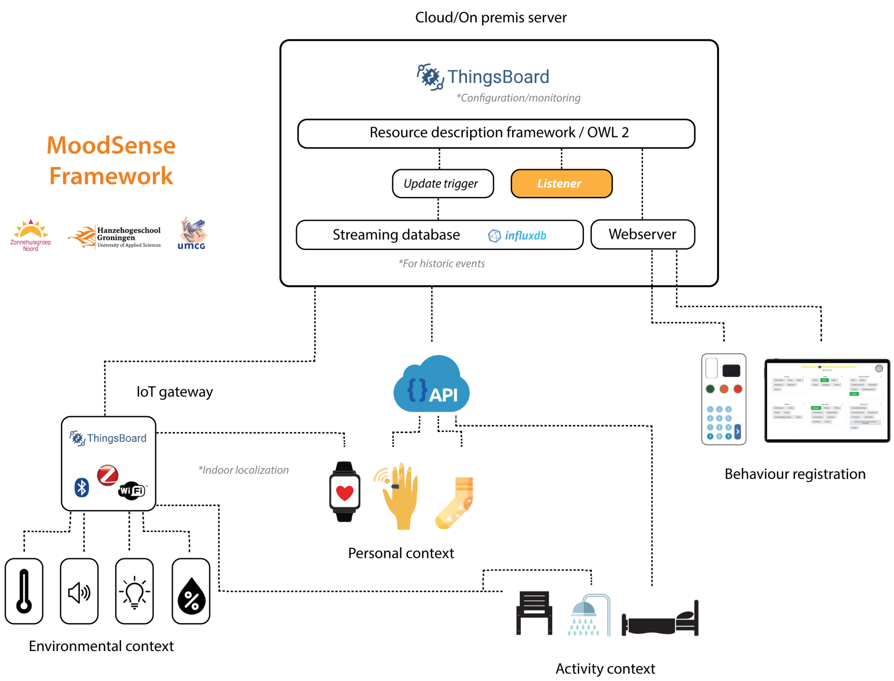

Figure 1.1: Current state of the MOOD-Sense initiative.

To address these challenges and create an energy-efficient network, the proposal to implement a Thread mesh wireless network was introduced. Thread's features, such as mesh networking, multiprotocol support, and low cost, make it an ideal solution for connecting BLE, ZigBee, and Wi-Fi connectivity together [4]. However, Thread devices can consume more power than other network types due to frequent activity needed for processing data from sensors or other end devices [5]. For instance, a sensor constantly monitoring a dementia patient's activity and sending data through the network at a very high frequency. This situation could lead to higher electricity costs and shorter device lifespans.

Optimizing the overall power consumption in the Thread network can save on electricity bills, extend device lifetimes, and enable the use of portable batteries to power the network when grid electricity isn't available. In settings like nursing homes where MOOD-Sense applications are deployed, reducing overall IoT network power consumption can significantly cut down electricity costs, making the system more cost-effective for care facilities. Additionally, this research aligns with sustainable research principles, reducing environmental impact by cutting down energy use. The research aims to create an energy-efficient and reliable Thread mesh wireless network for IoT applications like MOOD-Sense.

1.3 Desired Outcome

The desired outcome of this research is to develop an efficient algorithm that integrates seamlessly within the Thread-based wireless network system and optimizing power consumption. This algorithm will not only adjust transmission power but also select the most suitable device types for the network, contributing to energy efficiency and reliable communication. A schematic overview of the system with the integrated algorithm is as follows:

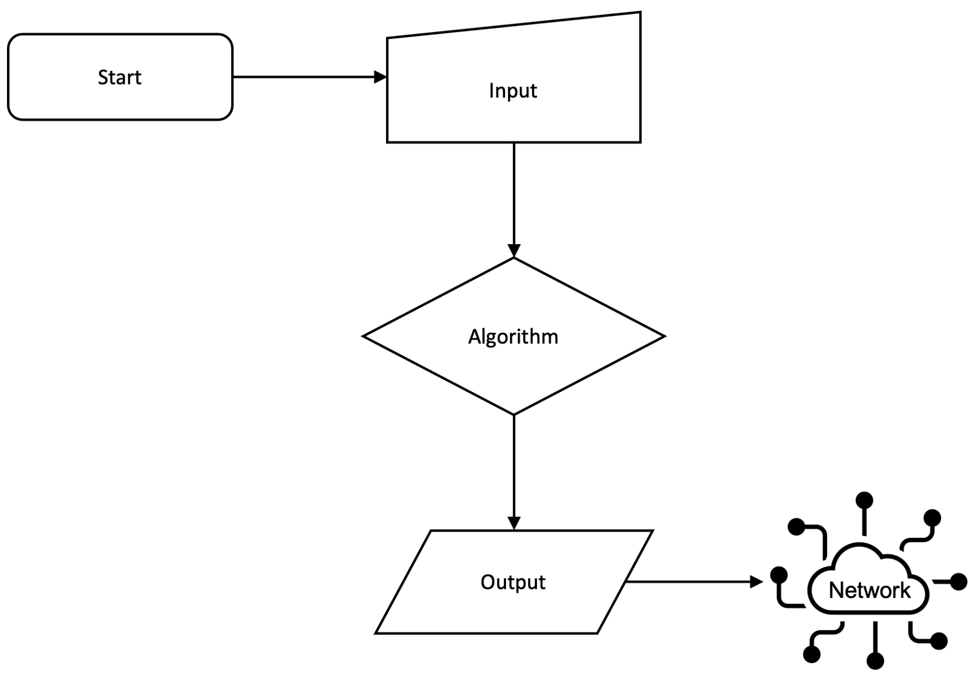

Figure 1.2: Schematic overview of desired outcomes.

1. **Input**: The primary input parameters for the network optimization algorithm include the total number of devices, the distance between each device, and network parameters. These inputs provide the necessary data to guide the optimization process and ensure that the algorithm makes informed decisions regarding device types and transmission power levels.

2. **Algorithm**: The power optimization algorithm consists of two stages: the Monte Carlo Method (MCM) and the Genetic Algorithm (GA). MCM and GA were chosen due to their ability to efficiently solve complex optimization problems that involve multiple variables and constraints. MCM is particularly effective in dealing with uncertainty and randomness in optimization problems, while GA offers a robust solution for finding optimal values in large search spaces [6]. In the first stage, MCM focuses on determining the right device types based on various constraints and constructing an optimal network configuration with an initial transmission power setting. The second stage involves GA, which takes the output from MCM and optimizes the transmission power settings to minimize power consumption while maintaining network reliability. One of GA's key strengths is its ability to avoid local optima and explore the search space more thoroughly, increasing the likelihood of finding global optima for the problem at hand [7].

3. **Output**: The output consists of the appropriate device types for a reliable Thread network configuration, along with the optimal transmission power settings for each device, and device positions. This output enables the creation of an energy-efficient and reliable Thread-based wireless network that meets the needs of the MOOD-Sense initiative and other similar IoT applications.

4. **Integration**: The output from the algorithm for optimal devices and roles within the network, such as border routers, routers, and end devices, will be integrated into the Thread network through a manual configuration process. Additionally, the optimal transmission power settings, as determined by the algorithm, will be applied to each device to minimize power consumption while maintaining network reliability. This manual configuration process ensures that the devices are set up for optimal performance based on the algorithm's recommendations. In the future, automated integration could be explored to streamline this process further.

By achieving this desired outcome, the algorithm will provide a comprehensive solution for power optimization in Thread networks, supporting the MOOD-Sense initiative and similar IoT applications in building energy-efficient and sustainable networks.

1.4 Problem Definition

As the adoption of IoT devices in applications like the MOOD-Sense initiative increases, there is a growing need for energy-efficient and reliable wireless network protocols. The Thread network protocol offers low-power and reliable mesh networking, making it suitable for such applications [3]. However, optimizing power consumption while maintaining network reliability remains a challenge. Additionally, the selection of appropriate device types is crucial for building an efficient Thread network, as Thread offers various device types depending on the use case.

The primary goal of this research is to determine the most effective algorithmic approach for power optimization in a Thread-based wireless network, specifically through transmission power adjustments and the selection of the appropriate device types. By focusing on these aspects, the research will contribute to the development of energy-efficient network solutions for the MOOD-Sense initiative and similar IoT applications. This approach ensures the proper selection and utilization of devices within the Thread network, optimizing the overall network performance and energy efficiency.

1.5 Main Research Question

How can parameter optimization be applied to develop a power-optimized Thread mesh wireless network?

1.6 List of Requirements

1. Optimize power efficiency for the Thread network protocol with a focus on minimizing power consumption while maintaining reliable communication, and assess the impact of location on power optimization performance for both maximum and optimized modes.
   Constraint: The optimization should not compromise the network's stability, communication quality, or applicability to diverse environments.

2. Employ MCM and GA for optimizing transmission power, determining efficient network configurations, investigating the significance of errors in the power optimization process, and evaluating their impact on the performance of MCM and GA modes.
   Constraint: The optimization techniques should be computationally feasible, not add significant overhead to the network's operation, and should identify potential sources of errors while recommending ways to minimize their impact on power optimization performance.

3. Develop a power-optimized Thread mesh wireless network by considering the optimal device types for different nodes, and compare the performance of MCM and GA modes across different device types and locations in terms of power optimization.
   Constraint: The selected device types should maintain low power consumption while meeting the network's performance requirements, and the comparison should be fair and unbiased.

4. Suggest future research directions and improvements for Thread network optimization, including device positioning, path loss, and broader application scope, while ensuring adherence to responsible research and innovation principles, including ethical aspects, professional skills, applied research, and sustainability.
   Constraint: The suggestions should be realistic and feasible, considering existing

limitations and challenges in the field, and the research should prioritize the development of sustainable solutions and maintain transparency and accountability throughout the process.

1.7 Sub-Research Questions

1. What are the key features of the Thread protocol that make it suitable for IoT applications, specifically in the context of the MOOD-Sense project, and what are the specific hardware requirements for implementing a Thread network?

2. Which parameters significantly impact the transmission power in a Thread network, and how do they relate to energy efficiency and network performance?

3. How do the MCM and the GA differ in their approach to optimizing transmission power in a Thread network, and what are the key steps for their implementation? How do variations in algorithmic parameters impact their performance?

4. What are the differences in power optimization performance between different iterations for both maximum and optimized modes, and how do various factors such as device performance, location, device types, and the correlation between mean, max, and min Current (mA) values impact the optimization process?

5. How do MCM and GA modes compare with maximum mode in terms of efficiency across various locations and device types, and what is the significance of errors in the power optimization process and their impact on the performance of MCM and GA modes?

Chapter 2

Situational & Theoretical Analysis

2.1 The Thread Protocol

Thread is a low-power, wireless IoT protocol designed to provide secure, reliable, and scalable networking for connected devices. Developed by the Thread Group, which includes notable members such as Nest Labs (a subsidiary of Google), Advanced RISC Machine (ARM), and Silicon Labs, Thread was introduced in 2014 to address the growing need for a standardized and efficient IoT networking solution. Built on open standards, Thread is an Internet Protocol version 6 (IPv6) based protocol that utilizes the Institute of Electrical and Electronics Engineers (IEEE) 802.15.4 radio standard for communication, making it compatible with a wide range of existing devices and technologies [3].

2.1.1 Architecture and Components

Thread's architecture is based on a mesh topology, allowing devices to communicate directly with each other, bypassing the need for a central hub or router. This mesh design enhances network resilience, as devices can automatically re-route communication through alternative paths if a connection is lost [3]. The key components of the Thread protocol include:

1. **Border Routers**: These devices serve as gateways between the Thread network and external Internet Protocol (IP) networks, such as Wi-Fi or Ethernet networks. They manage network access, security, and routing of data between the Thread network and other networks [3].

2. **Leader Routers**: They play a vital role in managing the network by assigning addresses to devices, coordinating routing updates, and maintaining overall network stability [3].

3. **Routers**: These devices are responsible for routing data within the Thread network. They can also act as parent devices to other devices within the network, providing connectivity to devices with limited routing capabilities [3].

4. **End Devices**: These devices communicate directly with their parent routers and are typically low-power devices, such as sensors or actuators. End devices do not participate in routing or network management [3].

5. **Links**: Thread networks use links to establish connections between devices, allowing them to communicate and exchange data. Links are essential for maintaining the mesh topology of Thread networks [3].

Figure 2.1 shows a visual representation of the basic Thread network topology, which includes all the listed components [3]. By combining these components, the Thread network architecture provides a robust, scalable, and energy-efficient solution for IoT applications, including the MOOD-Sense project.

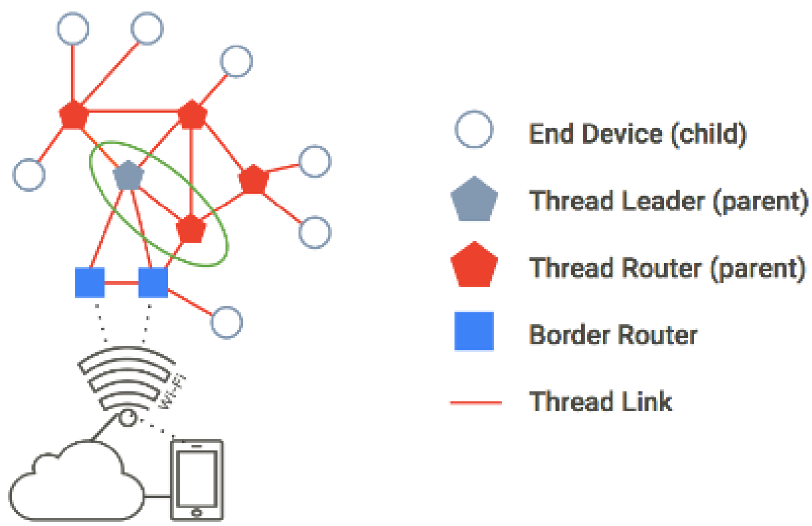

Figure 2.1: Basic Thread network topology [3].

2.1.2 Key Features and Advantages

Considering the specific requirements of various IoT applications, including the MOOD-Sense project, the following features and advantages of Thread make it a suitable choice:

1. **Low Power Consumption**: Thread's energy-efficient design aligns with the need for long battery life in devices that continuously monitor, collect, and transmit data in various IoT scenarios [3].

2. **Scalability**: The mesh topology of Thread networks allows for the seamless addition of new devices, enabling IoT projects to adapt and expand as needed [3].

3. **Security**: Thread's end-to-end encryption and secure commissioning processes ensure that communication between devices is protected, maintaining data privacy and security across diverse applications [3].

4. **Robustness and Reliability**: Thread's self-healing mesh network design ensures reliable and resilient communication, which is crucial for continuous monitoring and data collection in IoT applications [3].

5. **Interoperability**: Thread's open standards ensure compatibility with a wide range of devices and technologies, allowing IoT projects to integrate various sensors, devices, and communication technologies within a single, unified network [3].

Overall, the Thread protocol's features address the key question of its suitability for IoT applications in various contexts, including the MOOD-Sense project. Offering a secure, reliable, and energy-efficient networking solution, Thread meets the requirements for continuous monitoring, improved data collection, and seamless integration across industries.

2.2 Power Optimization

Optimizing power consumption in wireless IoT networks is a critical challenge, particularly for applications like the MOOD-Sense project, where devices are expected to operate for extended periods without frequent battery replacements or recharging. One effective approach to reduce power consumption is by minimizing transmission power while still maintaining reliable communication among devices, taking into account factors such as path loss and signal strength [8]. This section describes the implementation of transmission power control for Thread wireless networks, aiming to optimize the overall power consumption and ensure efficient and reliable communication among devices within the MOOD-Sense project context.

2.2.1 Factors Influencing Transmission Power

Parameters influencing transmission power in a Thread network are crucial for optimizing energy efficiency and network performance. By examining these factors, network designers can make informed decisions to achieve optimal performance under various conditions. A thorough understanding of these parameters is essential for implementing effective power management strategies and maintaining reliable communication within the network. Some of the factors include:

Distance

The distance between devices directly influences the transmission power, as a longer distance between devices typically results in higher path loss [9]. Therefore, devices that are farther apart may require higher transmission power levels to maintain a stable connection. The Euclidean distance matrix calculates the distance between pairs of devices [10]. The distance between devices i and j with coordinates $(x1, y1)$ and $(x2, y2)$ is calculated as follows:

$$distance\,(i,j) = \sqrt{(x^2 - x^1)^2 + (y^2 - y^1)^2} \tag{2.1}$$

This calculation helps account for spatial constraints and device placements, ensuring MCM random input generation considers these factors for a more efficient and optimized Thread network [10].

Received Signal Strength Indicator

Received Signal Strength Indicator (RSSI) is a measurement of the power level of a received radio signal. It helps to determine the link quality between devices in a wireless network. A higher RSSI value indicates a stronger received signal, which may require lower transmission power to maintain reliable communication [11]. The RSSI calculation, including transmit and receive antenna gains, is:

$$RSSI = P_t + G_t + G_r - L_p \tag{2.2}$$

Where P_t is the transmission power (dBm), G_t is the transmit antenna gain (dBi), G_r is receive antenna gain (dBi), and L_p is path loss (dB) [12].

Thread devices typically have an RSSI sensitivity of -100 dBm. This formula applies to uplink and downlink connections, offering a more accurate signal strength representation and aiding in network performance optimization and energy consumption [13].

Antenna Gain

The gain of the antennas used in the network can also affect the transmission power. A higher gain antenna can focus the radio signal more effectively, requiring less transmission power to achieve the same signal strength at the receiver [9].

Path Loss

Path loss refers to the attenuation of the radio signal as it propagates through the environment, depending on factors such as distance, frequency, and environmental conditions [14]. It significantly impacts the transmission power needed for reliable communication. In this research, the log-normal shadowing model is used for path loss calculations. To understand this model, the three key path loss models are discussed, providing insight into the principles and mathematical formulations involved in path loss calculations for power optimization.

1. **Free-Space Propagation Model**: The free-space propagation model is used for predicting the received signal strength in Line of Sight (LOS) environments, where there are no obstacles between the transmitter and receiver. It is often adopted for satellite communication systems. The Friis equation 2.3 describes the received power at distance d, considering non-isotropic antennas with transmit gain G_t and receive gain G_r [14]:

$$P_r(d) = \frac{P_t G_t G_r \lambda^2}{(4\pi)^2 d^2 L} \tag{2.3}$$

Where P_t represents the transmit power (w), d is the distance between transmitter and receiver (m), λ is the wavelength of radiation (m), G_t is transmit gain (dB), G_r receive gain (dB), and L is the system loss factor independent of the propagation environment. The free-space path loss $PL_F(d)$ can be directly derived without any system loss from equation 2.3:

$$PL_F(d)\,[dB] = 10log\left(\frac{P_t}{P_r}\right) = -10log\left(\frac{G_t G_r \lambda^2}{(4\pi)^2 d^2}\right) \tag{2.4}$$

Without antenna gains (i.e., $G_t = G_r = 1$), equation 2.4 is reduced to:

$$PL_F(d)\,[dB] = 10log\left(\frac{P_t}{P_r}\right) = 20log\left(\frac{4\pi d}{\lambda}\right) \tag{2.5}$$

2. **Log-Distance Path Loss Model**: The log-distance path loss model is a more generalized approach, accounting for the varying path loss exponent n depending on the environment. The path loss at distance d is given by equation 2.6, where d_0 is the reference distance at which the path loss inherits the characteristics of free-space loss [14]:

$$PL_{LD}(d)\,[dB] = PL_F(d_0) + 10nlog\left(\frac{d}{d_0}\right) \tag{2.6}$$

Where d_0 is a reference distance and n corresponds to free space which tends to change as shown in the following table.

Table 2.1: Path loss exponent for different environments.

Environment	Path Loss Exponent (n)
Free space	2
Urban area cellular radio	2.7 - 3.5
Shadowed urban cellular radio	3 - 5
In building line-of-sight	1.6 - 1.8
Obstructed in building	4 - 6
Obstructed in factories	2 - 3

The path loss exponent (n) varies based on the environment, as shown in table 2.1, and helps to adjust the log-distance path loss model for more accurate predictions. Lower values represent environments with fewer obstructions, such as free space, while higher values indicate more complex environments with buildings or other obstacles [14].

3. **Log-Normal Shadowing Model**: The log-normal shadowing model considers the random nature of shadowing effects, making it more suitable for realistic situations. The model is given by equation 2.7, where X_σ is a Gaussian random variable with a zero mean and a standard deviation of σ [14]:

$$PL\left(d\right)\left[dB\right] = \overline{PL}\left(d\right) + X_\sigma = PL_F\left(d_0\right) + 10nlog\left(\frac{d}{d_0}\right) + X_\sigma \tag{2.7}$$

In other words, this particular model allows the receiver at the same distance d to have a different path loss, which varies with the random shadowing effect X_σ [14].

Device Types

In a Thread network, devices can have different types and roles, such as routers, end devices, or border routers. These roles can impact the transmission power requirements, as routers may need to communicate with multiple neighboring devices, whereas end devices only need to communicate with their parent router [3].

2.2.2 Algorithmic Approaches

To optimize transmission power in a Thread network, algorithmic approaches can be employed. In this research, two algorithms are used to address transmission power optimization: the MCM and the GA. The MCM focuses on selecting the right device types and configuring an optimal network based on the constraints and requirements of the MOOD-Sense project. The GA, on the other hand, takes the output from the MCM and optimizes the transmission power settings for each device, ensuring minimal energy consumption while maintaining network performance.

Monte Carlo Method

The Monte Carlo Method (MCM) is a popular computational technique for simulating complex systems and estimating numerical results using random sampling. It is particularly useful for solving problems with a large number of variables and an extensive search space, where traditional analytical methods may be inefficient or infeasible [6]. In the context of this research, MCM is employed to generate and evaluate potential network configurations for optimizing power consumption. The key steps involved in the MCM are as follows:

1. **Problem Formulation**: Define the problem and its parameters, including the objective function, constraints, and variables [6]. For this research, the goal is to establish a reliable Thread network configuration with optimal device types and

roles while minimizing the initial power consumption of the entire network, ensuring network performance and reliability are maintained.

2. **Sampling**: Generate a random sample of potential solutions to the problem within the search space. This involves randomly selecting values for the variables within their specified ranges [6]. In the case of this research, the sample consists of different network configurations with various device types and roles, along with varying transmission power settings for each device in the network.

3. **Evaluation**: Assess the quality of each sampled solution using the objective function and constraints defined in the problem formulation [6]. In this research, the objective function is to achieve the optimal network configuration that ensures reliability and minimizes the initial power consumption of the entire network.

4. **Termination**: Repeat the sampling and evaluation steps for a predefined number of iterations or until a convergence criterion is met, such as reaching a desired level of accuracy or observing no significant improvement over a number of iterations [6].

Genetic Algorithm

The Genetic Algorithm (GA) is an optimization technique inspired by the process of natural selection. It uses a population of possible solutions and evolves them over time using genetic operators, such as mutation and crossover. The goal is to find an optimal solution for the given problem, such as transmission power in this research [7]. In a typical GA process, the following steps are executed:

1. **Initialization**: A population of candidate solutions is generated, either randomly or using a heuristic method. The population size is determined by a predefined parameter. Each individual in the population represents a potential solution to the problem [15]. In this research, the output from MCM is used as the initial population.

2. **Fitness Evaluation**: The quality of each candidate solution in the population is assessed using a fitness function, which quantifies how well the individual solves the problem. The fitness value provides an objective measure to compare and rank individuals within the population [15]. Each solution in the population is validated against the predefined constraints set forth in this research.

3. **Selection**: Based on their fitness values, individuals are selected for reproduction. Selection methods, such as tournament selection or roulette wheel selection, are employed to choose the fittest individuals, favoring those with higher fitness values to participate in the creation of offspring for the next generation [7]. For instance, consider the following example where two binary format chromosomes are selected:

   (a) **Chromosome 1**: 1101100100110110

(b) **Chromosome 2**: 1101111000011110

4. **Crossover**: This genetic operator combines the genetic information of two parent individuals to create offspring that inherit characteristics from both parents. Crossover promotes exploration of the search space and the exchange of beneficial traits between individuals [7]. Consider the following example:

   (a) **Parent Chromosome 1**: 11011 | 00100110110

   (b) **Parent Chromosome 2**: 11011 | 11000011110

   After performing crossover operation in random manner, the offspring are:

   (a) **Offspring 1**: 11011 | 11000011110

   (b) **Offspring 2**: 11011 | 00100110110

   In this case, the genetic information from the two parent chromosomes has been recombined to form the offspring chromosomes, which carry traits from both parents.

5. **Mutation**: Mutation is another genetic operator that introduces small random changes in the genetic information of individuals. This process helps maintain diversity within the population and prevents the algorithm from getting stuck in local optima [7]. For instance, the following binary chromosomes can undergo mutation, where selected bits are randomly switched from 0 to 1, or vice versa:

   (a) **Original Parent Chromosome 1**: 1101100100110110

   (b) **Original Parent Chromosome 2**: 1101111000011110

   After performing mutation operation in random manner, the offspring become:

   (a) **Mutated Offspring 1**: 1100111000011110

   (b) **Mutated Offspring 2**: 1101101000011010

   In this example, the original parent chromosomes have been mutated to create the new offspring chromosomes. The mutation operation has introduced new characteristics into the offspring, helping to maintain genetic diversity in the population.

6. **Replacement**: The least fit individuals in the population are replaced with the newly created offspring, ensuring the population size remains constant across generations [7].

7. **Termination**: The algorithm repeats the process of fitness evaluation, selection, crossover, mutation, and replacement for a predefined number of generations or until a convergence criterion is met, such as reaching a desired level of fitness or observing no significant improvement over a number of generations [7].

By combining MCM and GA, the proposed algorithmic approach efficiently explores the parameter space and identifies the optimal configuration that minimizes power consumption while maintaining network performance and reliability in the Thread network. Addressing the sub-research question 3, this research demonstrates the effectiveness of the combined approach in achieving the desired outcomes.

2.2.3 Step-by-Step Guide for Power Optimization

1. **Identify the Influencing Factors and Parameters**: Determine the parameters that impact transmission power, such as distance between devices, path loss, signal strength, interference, and device types.

2. **Develop the Algorithms**: Implement the MCM and GA algorithms to address power optimization in the Thread network, considering the constraints and requirements of the MOOD-Sense project.

3. **Determine Optimal Device Types and Network Configuration**: Use the MCM to identify the appropriate device types and optimal network configuration for the Thread network.

4. **Optimize Transmission Power Settings**: Apply the GA to optimize the transmission power settings for each device in the network, ensuring minimal energy consumption while maintaining reliable communication.

5. **Evaluate Network Performance**: Assess the overall performance of the Thread network, ensuring that the optimized power settings do not compromise the network's reliability or efficiency.

6. **Iterate and Refine**: Continuously refine the algorithms and network configuration as necessary to maintain optimal power consumption and network performance.

Through exploring the factors influencing transmission power and employing algorithmic approaches, this research seeks to provide a comprehensive understanding of power optimization in a Thread network and its implications on energy efficiency and network performance in the context of the MOOD-Sense project.

2.3 Hardware Analysis

In this research, several hardware components were employed to implement and optimize the Thread network for power efficiency. Each hardware piece played a crucial role in different stages of the project:

2.3.1 nRF52840 Development Kit

The nRF52840 DK is a versatile single-board Development Kit for Bluetooth 5, Bluetooth mesh, Thread, Zigbee, 802.15.4, Adaptive Network Topology (ANT), and 2.4 GHz proprietary applications on the nRF52840 System on Chip (SoC) [4]. In this research, the nRF52840 DK was used to develop and test the Thread network, acting as routers and end devices in the network topology. The Development Kit enabled the research team to implement and evaluate the network performance and power consumption under different configurations.

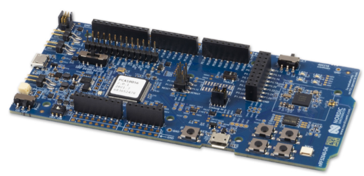

Figure 2.2: nRF52840 DK [4].

2.3.2 nRF52840 Dongle

The nRF52840 Dongle is a small, low-cost Universal Serial Bus (USB) device for the nRF Connect for Desktop Personal Computer (PC) tool. It supports Bluetooth 5, Bluetooth mesh, Thread, Zigbee, 802.15.4, ANT, and 2.4 GHz proprietary protocols [16]. In the research, the Dongle was used to extend the network by adding more nodes, facilitating the evaluation of scalability and network performance in larger network configurations.

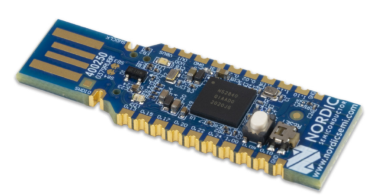

Figure 2.3: nRF52840 Dongle [16].

2.3.3 Power Profiler Kit II

The Power Profiler Kit (PPK) II is an easy-to-use tool for measuring and optimizing the power consumption of IoT devices [17]. In this research, the PPK II was utilized to measure the power consumption of the nRF52840 DK devices in various network configurations, enabling the research to assess the energy efficiency of the network and identify areas for improvement.

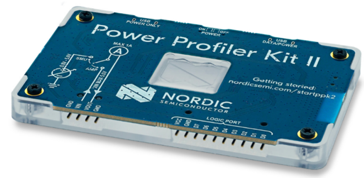

Figure 2.4: PPK II [17].

2.3.4 Raspberry Pi 4

The Raspberry Pi 4 model B is a single-board computer used for various applications, including IoT development [18]. In this research, the Raspberry Pi 4 served as a border router and provided an interface between the Thread network and external networks. The Raspberry Pi 4 allowed the research to evaluate the overall network performance and data exchange with external systems.

Figure 2.5: Raspberry Pi 4 model B [18].

By understanding the roles of each hardware component in the research, it becomes evident how they collectively contributed to the successful implementation and optimization of the Thread network for power efficiency.

2.3.5 Constraints and Limitations

Using these hardware components poses certain constraints and limitations on the research, with some potential consequences:

1. **Limited Scalability**: The number of available nRF52840 DK and nRF52840 Dongle devices may limit the size of the network being optimized, potentially affecting the generalizability of the results. This limitation might make it challenging to extrapolate the findings to larger networks or different device types.

2. **Hardware-Specific Performance**: The optimization results might be influenced by the specific hardware used, such as the nRF52840 DK and nRF52840 Dongle, and may not be directly applicable to other devices or platforms. As a consequence, further research and testing may be required to confirm the findings' applicability in different hardware contexts.

3. **Measurement Accuracy**: The accuracy of the PPK II may impact the precision of the power consumption measurements, potentially affecting the optimization results. This limitation could lead to underestimation or overestimation of energy savings, influencing the overall conclusions regarding the network's energy efficiency.

These hardware-related challenges could influence the research outcomes, making it essential to be aware of the limitations and consider their potential impact on the findings when interpreting the results and applying them to real-world scenarios.

2.3.6 Implications for Wireless Network Development

Taking into account the hardware constraints and limitations, the implications for wireless network development in the context of this research can be examined. The chosen hardware components, such as the nRF52840 DK, and nRF52840 Dongle directly impact the energy efficiency, performance, and scalability of the Thread network. Using these components enabled the investigation and optimization of power consumption and network performance. However, it is essential to acknowledge that hardware limitations might pose challenges when adapting the network to various IoT applications or scaling it to larger configurations. By addressing the sub-research question 1, this study highlights the importance of hardware selection and its implications for future wireless network technology development.

2.4 Literature Research

Thread network power consumption research has been limited but offers promising results. One study by Semiconductor [5] demonstrates that the battery life of a Thread node is heavily dependent on the network configuration. For example, a node with an idle current of 3 μA and a transmit current of 17 mA can last up to 10 years in a network with a low data rate of 250 $kbps$ and a small number of packets per day. However, in a network with a high data rate of 1 $Mbps$ and many packets per day, the same node would only last for a few months. A white paper by Group [19] provides noteworthy results on the power consumption and optimization of Thread networks, showing that the Thread protocol could achieve a standby power consumption of less than 3 mW, with typical transmit and receive power consumption ranging between 15 mW and 20 mW. The study also demonstrated that devices on a Thread network could achieve up to 10 years of battery life when transmitting once per minute, making Thread a strong candidate for low-power IoT applications. Another research effort, conducted by Azoidou, Pang, Liu, *et al.* [20] analyzed the power consumption of Thread end devices, routers, and coordinators. The study demonstrated that enabling power management features could reduce power consumption by up to 70% in sleep mode. Additionally, two power optimization techniques, dynamic power management and dynamic voltage and frequency scaling, were evaluated, with the latter having a greater impact, reducing consumption by up to 35%. The research also emphasized that power consumption is influenced by transmission power level, data rate, and routing topology and suggested that implementing optimization techniques could reduce power consumption by up to 70%.

In the paper by Sheth and Han [8] presents a practical implementation of transmit power control for 802.11b wireless networks. They focus on optimizing transmit power to reduce power consumption while maintaining correct reception of packets. The researchers achieved a maximum power savings of 25%, including idling power, by implementing their adaptive transmit power control algorithm as a user-level application layer process. This work is relevant to power optimization strategies in IoT applications, such as Thread networks. Behzad and Rubin [21] investigate the impact of transmission power on the throughput capacity of finite ad hoc wireless networks using a scheduling-based Media Access Control (MAC) protocol in their paper, such as Time Division Multiple Access (TDMA). The authors demonstrate that by properly increasing the nodal transmit power level, the capacity of an ad hoc wireless network can be maximized, regardless of nodal distribution and traffic pattern. The primary finding is that higher transmission power contributes to increased combinatorial diversity, optimizing joint scheduling and routing schemes, which is valuable for the development of efficient IoT applications using protocols like Thread.

In the realm of algorithm optimization, the MCM is a robust, efficient, flexible, and scalable tool used across various fields, including science, finance, and engineering. Research by Kroese, Brereton, Taimre, *et al.* [6] emphasizes MCM's popularity and its applications in areas like industrial engineering, operations research, physical processes, random graphs, finance, biology, medicine, and computer science. The authors highlight MCM's simplicity, strength in randomness, and theoretical justification. Girgis, Mahmoud, Abdullatif, *et al.* [22] propose a GA and Simulated Annealing (SA) for solving the Wireless Mesh Network (WMN) design problem. The study aims to minimize cost and determine the number of used gateways in WMN under constraints, with experimental results proving the effectiveness of GA and SA in minimizing network costs while satisfying quality of service. The authors find that GA outperforms SA in small-size networks, while SA performs better in large-size networks. On the other hand, GA is a heuristic optimization algorithm that handles non-linear, non-convex, and intermittent problems. It is widely applied in various engineering and scientific applications. One study by Ferentinos, Tsiligiridis, and Arvanitis [23] employs GA to optimize Wireless Sensor Networks (WSNs) for precision agriculture applications. The research determines active sensors, cluster heads, and signal ranges while considering network connectivity, energy conservation, and application requirements. Results indicate that GA-generated designs outperform random deployments regarding connectivity and energy consumption. Norouzi and Zaim [15] explore the potential of GA in optimizing the operational stages of WSNs, discussing node placement, network coverage, clustering, data aggregation, and routing. Simulations demonstrate that GA-based approaches outperform existing protocols, suggesting that GA can optimize WSNs in military, medical, and commercial applications.

In summary, although the literature on Thread power optimization is limited, the results from existing studies suggest that the protocol has significant potential for reducing energy consumption in low-power wireless networking applications. For instance, the research by Sheth and Han [8] demonstrates the effectiveness of optimizing transmit power to reduce power consumption, emphasizing the potential of using algorithmic approaches for power optimization in IoT applications such as Thread networks. Furthermore, the research by Girgis, Mahmoud, Abdullatif, *et al.* [22] shows that GA can effectively minimize network costs while satisfying quality of service, highlighting the potential for GA optimization in similar wireless network scenarios. Given the success of MCM and GA in other optimization scenarios, they were chosen to be explored within the context of Thread network power optimization, building upon the limited available literature and attempting to address the gaps in knowledge. This research aims to contribute valuable insights and drive further advancements in the field by applying the proven effectiveness of MCM and GA in power optimization, aligning with the findings of related studies and enhancing the potential for optimizing power consumption in low-power IoT applications.

Chapter 3

Conceptual Model

This section introduces the central theme of the research, which revolves around transmission power optimization in Thread mesh wireless networks as a part of the MOOD-Sense initiative. With a primary focus on improving energy efficiency in IoT applications utilizing the Thread protocol, the research delves into the investigation and evaluation of transmission power optimization with optimal network configuration using an algorithmic approach. This examination sheds light on the techniques' impact on overall network performance and contributes to the development of energy-efficient IoT networks.

The system will be developed based on the Thread protocol, a low-power, IPv6-based networking protocol designed for IoT applications. To optimize the power consumption of Thread networks, the project will employ a two-step process. First, the MCM will be used to find the optimal network configuration and initial transmission power. This step will involve a thorough analysis of different network configurations based on different constraints. The project will leverage MCM's strengths in randomness and theoretical justification to ensure the reliability of the results.

Next, the GA will take the final output from MCM and focus on finding the lowest transmission power possible. The use of GA will help improve the overall energy efficiency and performance of the Thread network by taking into account the network's constraints. The following diagram illustrates the flow of the entire process, from MCM and GA optimization to the implementation of optimized transmission power in the Thread network that shows a clear visual representation of the project's methodology.

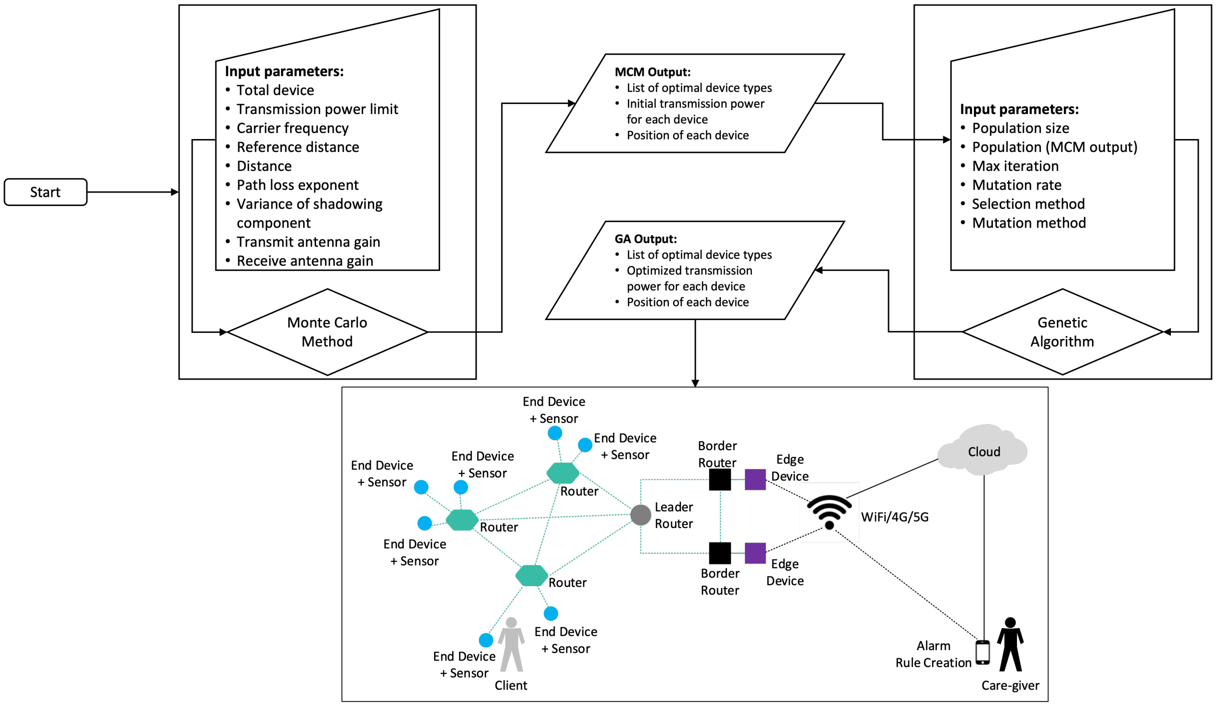

Figure 3.1: Thread network power optimization conceptual model.

The project will consider the cost of hardware components, software development, testing, and deployment while maintaining a balance between cost-effectiveness and performance. It is important to take into account the computational time and hardware requirements when implementing the MCM and GA. The MCM generally provides an initial solution more quickly, while the GA refines this solution and converges to an optimal one over a longer period due to its iterative nature and the use of genetic operators like crossover and mutation. The hardware requirements for both methods depend on the complexity of the problem and the size of the search space. However, modern computational resources are typically sufficient to handle the demands of these algorithms for the given research problem. Ultimately, this research emphasizes the importance of balancing the algorithmic approach with the underlying computational resources when optimizing power consumption in Thread networks.

The power consumption will be measured using PPK II, as explained in the hardware section. The research will measure output power in different scenarios to validate the effectiveness of the power optimization techniques employed. These scenarios will be categorized based on the method, location, type, mode, duration, and ping used for power optimization and measurement.

1. **Method**: The power consumption will be measured in two primary scenarios - Maximum and Optimized. The maximum scenario represents the baseline power consumption, where no optimization techniques are applied. The Optimized scenario will measure power consumption after implementing the MCM and GA optimization techniques.

2. **Location**: The measurements will be conducted in two different locations - Lab and Home. The lab setting is smaller in size compared to the home location, allowing for controlled environments and reproducible results. The home setting provides a real-world context, with a larger area, helping to understand the performance of the Thread network in everyday IoT applications.

3. **Type**: The power consumption measurements will also be conducted based on the type of network activity. The no sensor scenario represents a Thread network with no active sensors, while the ping scenario simulates data exchange between nodes, resembling real IoT network behavior.

4. **Mode**: The project will compare the effectiveness of MCM and GA optimization techniques. The MCM mode will measure power consumption based on network configurations optimized using the MCM. The GA mode will measure power consumption with network configurations optimized using the GA.

5. **Duration**: The power consumption measurements will be conducted for different durations - 60 seconds in the lab location and 300 seconds in the home location. This variation in duration will help in understanding the impact of time on power consumption in different environments.

6. **Ping**: In the lab location, 50 pings will be sent within the 60-second duration, whereas in the home location, 290 pings will be sent during the 300-second duration. This distinction will help analyze the impact of network activity on power consumption in both controlled and real-world settings.

By measuring power consumption in these different scenarios, the research will provide a comprehensive understanding of the power optimization techniques' effectiveness. The results will be analyzed to draw comparisons and determine the optimal approach for power consumption reduction in Thread networks, ultimately contributing to the development of energy-efficient IoT networks.

In terms of sustainability, the research will emphasize energy-efficient hardware and power optimization techniques to minimize environmental impact, leading to sustainable IoT network deployments. By focusing on energy efficiency, the project inherently follows sustainable work principles, addressing resource conservation and minimizing negative impacts on society and the environment. Moreover, the research process involves the application and development of professional skills, such as data analysis, algorithm design, and critical thinking, to ensure the reliability and relevance of the results.

By combining the use of MCM and GA to optimize power consumption in Thread networks and employing sustainable work principles, this project will contribute valuable insights to the field of energy-efficient IoT network design and implementation.

Chapter 4

Research Design

4.1 Mathematical Constraints

The objective of the mathematical model is to build a Thread network that adheres to specific mathematical constraints, ensuring a well-functioning network with the optimum device types, sensitivity, and RSSI. By complying with the constraints, the Thread network can be effectively optimized for both performance and energy consumption. The constraints of the model are as follows:

1. To establish a link between the devices within the network, the RSSI of each device must be approximately above the sensitivity of the device, which is -100 dBm with IEEE 802.15.4 [4]. This ensures a stable connection between the devices [9].

2. The transmission power limitation for all types of devices, ranging from -20 dBm to 8 dBm, is set according to the hardware specifications of the devices in the Thread network, ensuring optimal performance while facilitating power optimization techniques within these constraints for energy efficiency [13].

3. The number of Router Eligible End Devices (REEDs) must be equal to the number of routers and the leader because, if a router is lost, a connected REED must become a router to replace the dead router and maintain network resilience as part of the Thread self-healing feature [3].

4. Sleepy End Devices (SEDs) are end devices that save energy by entering a low-power sleep mode when not actively communicating. SEDs can be present or absent in a network, but their inclusion helps optimize power consumption due to their energy-efficient sleep periods [3].

5. In Thread networks, a leader router is always present, automatically elected through a decentralized process. This router manages network-wide configurations and operations, ensuring consistent performance and stability. Its constant presence supports the smooth functioning of the Thread network, adapting to changes in network topology or router failures [3].

6. In a mesh network, at least two routers are required to establish connectivity. However, having three or more routers, including a leader, greatly improves the network's resilience, redundancy, and coverage. Therefore, to create a more robust mesh network with a leader, it is recommended to have a minimum of three routers, with one of them serving as the leader. This configuration ensures enhanced network performance and maintains seamless communication throughout the network [22].

7. Thread networks can have multiple border routers, with at least two present to prevent a single point of failure. This redundancy ensures continuous connectivity and communication between the Thread network and other IP-based networks, maintaining network stability and reliability even if one border router encounters a failure [3].

The following mathematical model is designed for this purpose [22]:

$$Min \sum_{i=1}^{M} P_{tx}^{i} \tag{4.1}$$

Subjects to:

$$\begin{aligned} RSSI_{Device}^{j} > Sensitivity, j \in 1, \cdots, N \\ -20dBm \leq P_{t}^{j} \leq 8dBm \\ N_{REED} = N_{Router} + N_{Leader} \\ SED \in 0, 1 \\ N_{Leader} = 1 \\ N_{Router} + N_{Leader} \geq 3 \\ N_{BR} = 2 \end{aligned} \tag{4.2}$$

Where P_{tx} represents the transmit power (dBm) of each one of the M devices and N is amount of devices.

4.2 Monte Carlo Method Process

The MCM involves four main steps. First, the process is initialized with predefined parameters and constraints. Second, random numbers are generated within the defined bounds to explore various network configurations. Third, the generated configurations are evaluated based on their performance and adherence to constraints. Finally, after a predetermined number of iterations or reaching an acceptable solution, the MCM process comes to an end, providing an optimized network configuration. For a detailed explanation of each step, refer to the respective sections below.

4.2.1 Initialize

The MCM is initiated to optimize the Thread network, considering the key parameters influencing the network's performance and energy efficiency. These parameters are outlined in the table below:

Table 4.1: Parameters influencing Monte Carlo Method.

Param	Description
N_d	The total number of devices participating in the network, which is set to 8 for this research, representing a small-scale IoT network.
P_{tx}	Determines the signal strength for each device, randomly generated in a range between -20 dBm and 8 dBm according to the mathematical constraints, affecting network connectivity and energy consumption [13].
F_c	The carrier frequency used for calculating RSSI using the general path loss model, set at 2.4 GH_z, based on Thread protocol specification [3].
D_0	A reference distance of 0.25 m, associated with the carrier frequency F_c, employed in the log-distance path loss model in equation 2.6 to calculate the signal attenuation [14].
d	Represents the distance between two devices in the network, as illustrated in figures 4.3 and 4.4, influencing the strength of the signal received by devices.
n	The path loss exponent shown in equation 2.6, set to 5.0, which represents the rate at which the signal power decays with distance in the path loss model [14].
σ	The variance of the shadowing component, set to 3.0 dB, accounts for signal fluctuations due to obstacles and multipath propagation in the environment as shown in equation 2.7 [14].
G_t	The transmit antenna gain, set to 0.0 dB, which reflects the effectiveness of the transmitting antenna in directing the radio waves towards the receiving device [13].
G_r	The receive antenna gain, set to 0.0 dB, indicating the receiving antenna's ability to capture incoming radio waves [13].

4.2.2 Generate Random Numbers

Based on the factors mentioned at the start, MCM generates a vector X of length equal to $2n$, where n is the number of places where network elements can be allocated [21]. The vector is represented as:

$$X = [x_1, x_2, x_3, \cdots x_n, p_1, p_2, p_3, \cdots, p_n]$$
$$for\ x_n \in 0, 1, 2, 3, 4, 5 \tag{4.3}$$
$$p_n \in -20 : 4 : 8\ dBm$$

Where 0 represents no element allocated, 1 is allocate as a SED, 2 is allocate a REED, 3 is allocate a router, 4 is allocate the leader, and 5 is allocate a border router.

4.2.3 Evaluate Results

The objective function aims to build a Thread network using the optimal network configuration without violating the mathematical constraints. If a constraint is violated, a penalty is added to the objective function, which is weighted according to the importance of the constraint. The objective function with penalty values can be written as:

$$Min \sum_{i=1}^{M} P_{tx}^{i} + penal_1 + penal_2 + penal_3 + penal_{nr} \tag{4.4}$$

Where $penal_1$ represents penalty for violating the first restriction, $penal_2$ is penalty for violating the second restriction, and $penal_{nr}$ is the penalty for violating the last restriction.

4.2.4 Termination

The MCM converges on an optimal solution that satisfies necessary constraints, providing outputs such as device types, transmission power, and position. It also offers information on constraint violations, including the penalty, power consumption, and RSSI sensitivity violations—these outputs aid in understanding the optimization process and refining the network design. For a comprehensive understanding of the four steps of the MCM process, refer to the following pseudocode, which provides an overview of the algorithm's structure and logic.

Algorithm 1 MCM pseudocode for network optimization.

```
Initialize MCM parameters: N_d, d, F_c, D_0, n, σ, G_t, G_r
while network do
    devices, txpower, position ← generate_random_numbers(N_d)
    penalty, path_loss, rssi ← mathematical_constraints_evaluation(N_d, d, F_c, D_0, n,
    σ, G_t, G_r)
    if penalty is False then
        network ← False
    end if
    return devices, txpower, penalty, path_loss, rssi
end while
```

It is a simplified version of the MCM implementation and does not cover all the details of the original code. It is focused on the primary structure and steps of the method for network optimization and initial network build-up transmission power. To access the complete version of the algorithm code, including all implementation details, refer to the appendix 6.2 section.

4.3 Genetic Algorithm Process

The GA process can be summarized into four main steps: initializing population, evaluating fitness, performing selections, and finding the best solution. These steps are designed to optimize transmission power in the network by evolving a population of candidate solutions through generations. In the following paragraphs, each step is discussed in detail.

4.3.1 Initialize

The initial steps of the GA process start with creating a random population with the specified population size, representing different possible network configurations. The population is generated based on the parameters set, as shown below:

Table 4.2: Parameters influencing GA.

Param	Description
Population size	The number of individuals in the population representing different possible network configurations are set to 100 for this research.

Population	An initial random population is created with the specified population size and MCM output, which includes device types, transmission power, and device positions, representing different possible network configurations. For instance: [[device: 3, 5, 2, 5, 1, 5, 0, 0], [position: 1, 2, 3, 4, 5, 6, 7, 8], [txpower: -20, 0, 0, -8, 0, -12, 0, -20]].
Max iteration	The maximum number of iterations to be performed by the GA, for instance, 100 in this research.
Mutation rate	The probability of mutation is set at 0.1 for this research, determining the frequency of random changes introduced to the offspring's genetic information during the optimization process, which helps maintain genetic diversity within the population.
Selection method	The method used for selecting individuals from the current population to create the next generation, such as roulette wheel selection, tournament, or sorted. In this research, the sorted selection method was utilized.
Mutation method	The method used for mutating individuals affect how genetic information is altered during the mutation process. In this research, the swap mutation method was utilized.

4.3.2 Evaluate Fitness

In the fitness evaluation stage of the GA, each individual in the population is evaluated based on a fitness function. This fitness function is responsible for computing the fitness score of each individual, which, in this context, is represented as a chromosome [7]. Each chromosome in the population is composed of a list representing different device types, their respective positions, transmission powers, and an initially assigned penalty score of zero. For instance, here are some examples of chromosomes in the population:

1. **Chromosome 1**: [[device: 2, 5, 3, 3, 5, 2, 2, 4], [position: 1, 2, 3, 4, 5, 6, 7, 8], [txpower: -8, 8, 0, -16, 0, -8, -20, 8], [penalty: 0]]

2. **Chromosome 2**: [[device: 2, 5, 3, 3, 5, 2, 2, 4], [position: 1, 2, 3, 4, 5, 6, 7, 8], [txpower: -12, 4, -8, -8, -20, 0, -8, -20], [penalty: 0]]

3. **Chromosome 3**: [[device: 2, 5, 3, 3, 5, 2, 2, 4], [position: 1, 2, 3, 4, 5, 6, 7, 8], [txpower: 4, -12, -20, -4, -8, -12, -20, -8], [penalty: 0]]

For each chromosome, the fitness function calculates the path loss and RSSI sensitivity for each device configuration. The fitness function ensures that each chromosome adheres to the established mathematical constraints. When a chromosome violates a constraint, a penalty is added to the penalty score of that chromosome. The calculation and addition of penalties occur after the evaluation of each constraint, adjusting the fitness value accordingly. The fitness value, consequently, provides a measure of the quality of each solution, with lower penalties indicating more desirable solutions.

4.3.3 Selection

In the selection process, the sorted method is employed to identify the fittest individuals in the current population [7]. The entire population is sorted based on their penalties for constraint violations. The most fit individuals are then selected and stored in a separate list. Let's consider the following three chromosomes excluding device types and positions:

1. **Chromosome 1**: [[txpower: -8, 8, 0, -16, 0, -8, -20, 8], [penalty: 0]]
2. **Chromosome 2**: [[txpower: -12, 4, -8, -8, -20, 0, -8, -20], [penalty: 4000]]
3. **Chromosome 3**: [[txpower: 4, -12, -20, -4, -8, -12, -20, -8], [penalty: 3000]]

The sorted list of chromosomes will look like this:

1. **Chromosome 1**: [[txpower: -8, 8, 0, -16, 0, -8, -20, 8], [penalty: 0]]
2. **Chromosome 3**: [[txpower: 4, -12, -20, -4, -8, -12, -20, -8], [penalty: 3000]]
3. **Chromosome 2**: [[txpower: -12, 4, -8, -8, -20, 0, -8, -20], [penalty: 4000]]

Chromosome 1, having the lowest penalty, is the fittest chromosome. This approach ensures that the algorithm focuses on the most promising solutions as it progresses through the crossover and mutation stages.

4.3.4 Crossover

The crossover operation utilizes genetic material from the output of the selection method to form two parent solutions, subsequently creating offspring that inherit properties from both parents. This process aims to explore new potential solutions in the search space by allowing offspring to possess a mix of characteristics from their parents [7]. For instance, consider the following parent chromosomes selected from the previous selection process:

1. **Chromosome 1**: [[txpower: -8, 8, 0, -16, 0, -8, -20, 8], [penalty: 0]]
2. **Chromosome 2**: [[txpower: 4, -12, -20, -4, -8, -12, -20, -8], [penalty: 3000]]

A random crossover point is determined within the length of the parent solutions, say at the fourth index for this instance. Consequently, offspring are generated by merging the first part of Parent Chromosome 1 up to the crossover point with the second part of Parent Chromosome 2 from the crossover point onwards, and vice versa. This results in the following offspring:

1. **Chromosome 1**: [[txpower: -8, 8, 0, -16, -8, -12, -20, -8], [penalty: TBD]]

2. **Chromosome 2**: [[txpower: 4, -12, -20, -4, 0, -8, -20, 8], [penalty: TBD]]

Here, "TBD" indicates that the penalty for each offspring chromosome will be determined in subsequent processes based on the updated txpower values.

Through this method, two new offspring are produced, each inheriting distinct portions of the parent solutions. This mechanism potentially leads to enhanced solutions in subsequent generations, thereby ensuring the evolution of the population towards optimal solutions.

4.3.5 Mutation

The mutation stage further enhances the diversity within the population, ensuring a thorough exploration of the search space. Offspring from the crossover stage serve as the input to the mutation operation [7]. For instance, consider the following offspring solutions:

1. **Chromosome 1**: [[txpower: -8, 8, 0, -16, -8, -12, -20, -8], [penalty: TBD]]

2. **Chromosome 2**: [[txpower: 4, -12, -20, -4, 0, -8, -20, 8], [penalty: TBD]]

A mutation rate controls the probability of mutation for each element within the offspring solution. If a random value, obtained through a uniform distribution, is less than the mutation rate, the respective element undergoes mutation [7]. In this instance, a mutation could be a change in the txpower value. Let's say the fourth element of Offspring Chromosome 1 and the second element of Offspring Chromosome 2 are selected for mutation. The txpower values may then be adjusted, resulting in the following mutated offspring:

1. **Chromosome 1**: [[txpower: -8, 8, 0, -10, -8, -12, -20, -8], [penalty: TBD]]

2. **Chromosome 2**: [[txpower: 4, -10, -20, -4, 0, -8, -20, 8], [penalty: TBD]]

Here, "TBD" also implies that the penalty for each mutated offspring chromosome will be recalculated in subsequent processes based on the new txpower values.

This mutation process ensures that the offspring solutions can investigate different combinations of transmission powers, which could potentially yield superior overall solutions in future generations.

4.3.6 Population Update

The algorithm iteratively refines the population by applying the selection, crossover, and mutation operations in each generation. After the offspring are created through crossover and mutation, their fitness values are calculated again. The population is then updated by replacing the current individuals with the new offspring, sorted based on their fitness values. This process of updating the population ensures that the best solutions are carried forward to the next generation [7].

4.3.7 Termination

The algorithm continues this process of generating new offspring and updating the population for a specified number of iterations. Once the termination criterion is met, the algorithm concludes, and the final population along with their fitness values are returned as output [7]. The best solution can be extracted from this final population, representing the optimal device types, optimized transmission power, and positions for each device in the given problem scenario. The following pseudocode gives an overview of the Genetic Algorithmprocess:

Algorithm 2 GA pseudocode for transmission power optimization.

```
Initialize GA parameters: population_size, population, max_iterations, mutation_rate, selection_method
Initialize population: create_random_population(population_size)
for each candidate in population do
    fitness = evaluate_fitness(candidate)
end for
for generation in range(max_iterations) do
    parents = select_parents(population, selection_method)
    offspring = crossover(parents, crossover_prob)
    offspring = mutate(offspring, mutation_prob)
    for each candidate in offspring do
        fitness = evaluate_fitness(candidate)
    end for
    population = replace_population(population, offspring)
end for
best_solution = find_best_solution(population)
```

The output is a list of optimized transmission power values for each device, along with optimal device types, and positions. For a comprehensive understanding of the Genetic Algorithm's implementation, refer to the appendix 6.2 for the complete code.

4.4 Experimental Setup

The prototype was built to validate the output from MCM and GA, using the optimal network configuration determined by MCM, which consisted of a total of 8 devices. The setup included 2 border routers, 3 routers (with one of them automatically elected as a leader), and 3 REEDs. The prototype was designed to closely resemble the conceptual model presented earlier in the figure, with the only slight difference being the use of REEDs instead of sensors as the end devices. An image was provided below to illustrate the Thread network topology that had been constructed.

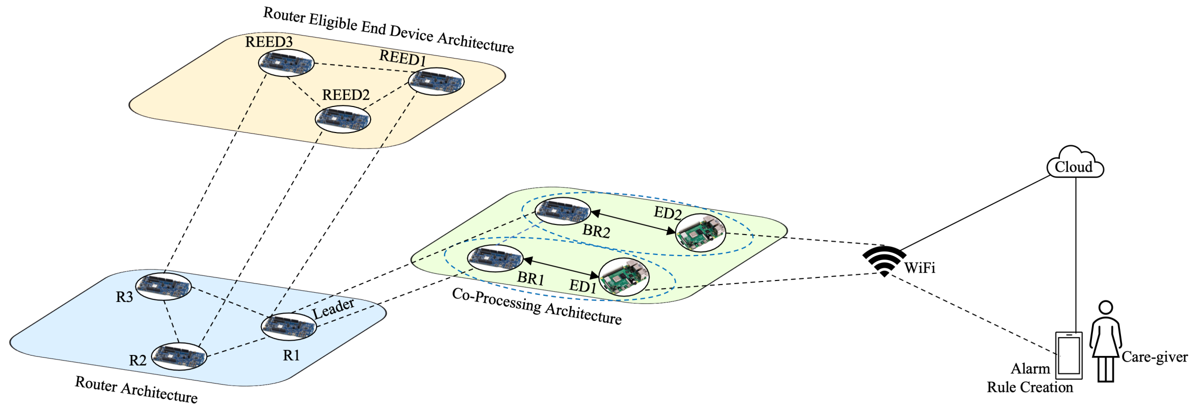

Figure 4.1: Thread network topology of the prototype.

The construction process of the prototype involved several crucial steps, aimed at validating the output from MCM and GA and ensuring optimal network configuration:

1. Customized nRF Thread Client and Server Software Development Kit (SDK) to fit the needs for the research, selecting roles for each device and setting the transmission power output from both MCM and GA for optimal network configuration.

2. Flashed each router with the Thread Server and each REED with the Thread Client customized SDK. In this configuration, routers acted as servers, while REEDs acted as clients. Communication between devices was bidirectional, with the clients having BLE enabled for multiprotocol support.

3. Flashed the border router nodes with the Coprocessor setup provided by nRF. To enable the Raspberry Pi to act as an edge device, the OpenThread Radio Coprocessor (RCP) architecture was implemented.

4. Turned on the devices one by one, noting that the first device activated in the network was most likely to become the leader, although leadership could change during the network's lifetime.

5. Validated all the nodes by running multicast messages using Thread Internet Control Message Protocol (ICMP) service. The ICMP service allowed sending echo requests (ping) to devices, activating their Thread antennas. This enabled testing the Thread connection, and devices could also reply [3].

6. Validated the multiprotocol support connection by running a data flow from the ESP32 Ultra-Wide Band (UWB) and mobile devices to the REEDs through BLE, which then forwarded the data to the routers. This step ensured seamless communication between non-Thread devices and the Thread network.

7. Monitored the network for stability and performance, adjusting settings to maintain optimal operation.

Following these steps, the prototype was successfully constructed to apply the optimized settings obtained from the MCM and GA. The subsequent figure presented a real-world Thread network prototype setup from the lab setup. The image provided a clear view of the nRF52840-based Thread nodes, Raspberry Pi as the edge device, and the border router setup with the dongle. It also showcased the development kits used for routers and REEDs.

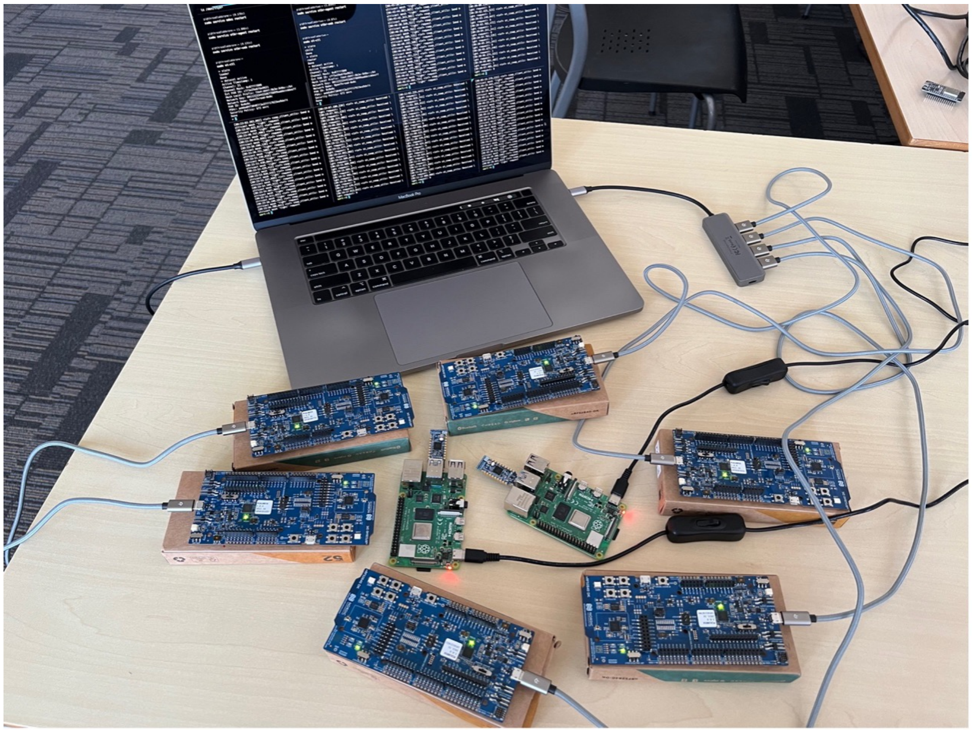

Figure 4.2: Thread network prototype setup in the lab.

4.5 Data Collection

The data collection process aimed to validate and compare the solutions from MCM and GA against the maximum mode by measuring power consumption in each device of the built prototype. Utilizing the nRF PPK II, which offers Current (*ma*) measurement at 100,000 samples per second, allowed for accurate power consumption measurements across various scenarios, locations, network activities, optimization modes, and durations. This approach provided insights into the effectiveness of the optimization techniques in both controlled and real-world settings while avoiding excessive data that would have complicated the analysis process.

1. **Method**: Power consumption was measured in two primary scenarios - Maximum

and Optimized. The maximum scenario represented the baseline power consumption, where no optimization techniques were applied. The optimized scenario measured power consumption after implementing the MCM and GA optimization techniques.

2. **Location**: Measurements were conducted in two different locations - Lab and Home. The lab setting, smaller in size compared to the home location, allowed for controlled environments and reproducible results. The home setting provided a real-world context with a larger area, helping to understand the performance of the Thread network in everyday IoT applications. The following images show the Euclidean distance matrix from two different locations to share a clear view of the distance between each device in the two locations.

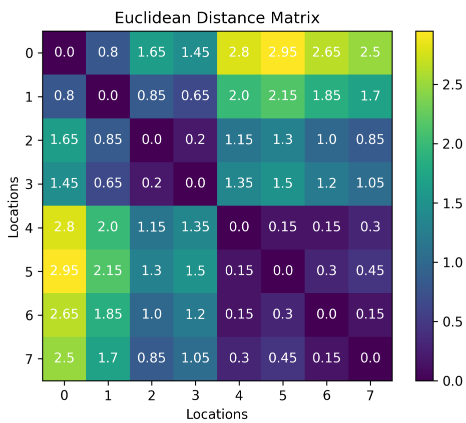

Figure 4.3: Distance matrix for lab.

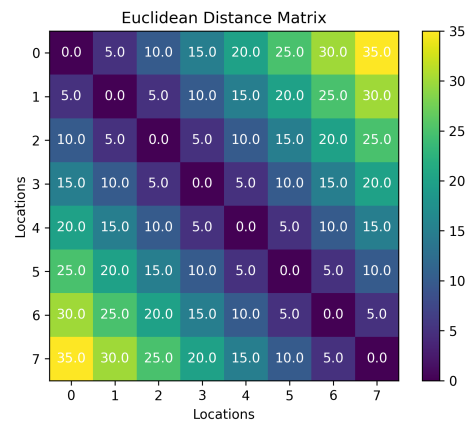

Figure 4.4: Distance matrix for home.

3. **Type**: Power consumption measurements were also conducted based on the type of network activity. The no sensor scenario represented a Thread network with no active sensors, while the ping scenario simulated data exchange between nodes, resembling real IoT network behavior [3].

4. **Mode**: The project compared the effectiveness of MCM and GA optimization techniques. The MCM mode measured power consumption based on network configurations optimized using the Monte Carlo Method. The GA mode measured power consumption with network configurations optimized using the Genetic Algorithm.

5. **Duration**: Power consumption measurements were conducted for different durations - 60 seconds in the lab location and 300 seconds in the home location. This variation in duration helped in understanding the impact of time on power consumption in different environments.

6. **Ping**: In the lab location, 50 pings were sent within the 60-second duration, whereas in the home location, 290 pings were sent during the 300-second duration. This

distinction helped analyze the impact of network activity on power consumption in both controlled and real-world settings.

Following the data collection steps, two images are provided to illustrate the process of collecting power consumption data from the nRF52840 DK using the PPK II. These images offer a visual representation of the setup and the data collection process, giving a clearer understanding of the experimental context and the methods used for obtaining the power consumption measurements.

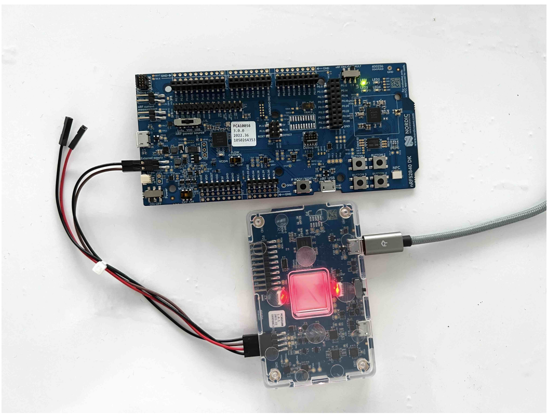

Figure 4.5: PPK II connected to a router.

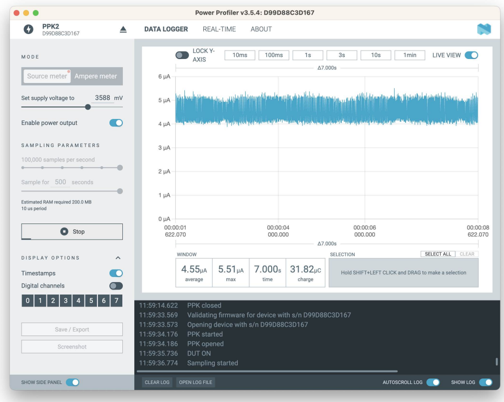

Figure 4.6: PPK II software in source meter mode.

Chapter 5

Research Results

5.1 Monte Carlo Method Analysis

The MCM analysis focuses on the output derived from two distinct locations: the Lab and Home. Due to the differences in size between these locations, the distances between devices as input parameter vary, as illustrated in figures 4.3 and 4.4. In the following tables, only the last 5 iterations of the MCM output are presented, as space limitations in this research paper prevent the inclusion of all iterations, which can number in the thousands. For a comprehensive list of the data table, refer to the appendix. The full list of parameters used for the MCM analysis can be found in table 4.1.

Table 5.1: MCM output from lab.

Device	$P_{tx}(dBm)$	Penalty
3, 5, 2, 5, 1, 5, 0, 0	-20, 0, 0, -8, 0, -12, 0, -20	3000
3, 4, 1, 3, 0, 4, 1, 3	-16, -8, -4, -20, 0, -12, -4, -20	3000
3, 4, 4, 1, 2, 4, 4, 2	4, -12, -20, -4, -8, -12, -20, -8	3000
2, 0, 1, 2, 0, 0, 1, 1	-12, 4, -8, -8, -20, 0, -8, -20	4000
2, 5, 3, 3, 5, 2, 2, 4	-8, 8, 0, -16, 0, -8, -20, 8	0

Table 5.2: MCM output from home.

Device	$P_{tx}(dBm)$	Penalty
2, 0, 2, 2, 5, 5, 1, 1	-12, -8, 8, -4, 0, 8, -20, -12	3000
4, 1, 1, 4, 0, 5, 1, 0	-12, 4, 0, -8, 8, 8, -12, -20	4000
0, 2, 5, 2, 4, 2, 5, 4	-20, -4, -12, -12, -4, -8, -20, -4	3000
1, 2, 0, 2, 1, 2, 0, 1	-8, -8, -16, -16, -12, -8, -16, 4	4000
2, 3, 5, 2, 2, 3, 4, 5	8, 8, -20, -8, -4, -16, 4, -16	0

The last row in each table indicates a penalty value of 0, which satisfies the mathematical constraints. When a constraint violation occurs, a penalty value of 1000 is added to the penalty column. An optimal network configuration, which comprises different device types and initial transmission power, is represented by the absence of a penalty. Table 5.1 shows the MCM output from the lab location, where the final row demonstrates an optimal network configuration, with a penalty value of zero. Similarly, table 5.2 presents the MCM output from the home location, with the last row indicating an optimal configuration, also featuring a penalty value of 0.

Upon analyzing the rows with penalties in both tables, the first row can be considered as an example. In this row, the penalty value is 3000, which signifies three violations. As per the mathematical constraints, a leader router must be present in the network, denoted by the number 4 in the device column. The lack of a leader router leads to the first violation, contributing 1000 to the penalty. The associated mathematical constraints and models are elaborated in equations 4.1, 4.2, 4.2, and 4.2.

The next constraint requires the number of routers and leaders to be equal to or greater than 3. However, the network configuration in the first row lacks a leader, leading to another violation. Lastly, a constraint mandates that the number of REEDs must be equal to the combined number of routers and leaders. The absence of a leader router in the network configuration causes the penalty value to reach 3000. The MCM continues iterating until it identifies an optimal network configuration without any constraint violations.

5.2 Genetic Algorithm Analysis

Similar to the MCM analysis, the tables presented below display the output from the Genetic Algorithm for both lab and home locations, with the primary difference between the two scenarios being the distance between devices as input parameters. Unlike the MCM, GA directly provides the final result, showcasing the lowest feasible transmission power without any constraint violations. As GA emphasizes minimizing transmission power, storing all analyzed data from its output is unnecessary, except for the final result. The full list of parameters used in the GA analysis can be found in table 4.2.

Table 5.3: GA output from lab.

Device	$P_{tx}(dBm)$	Total
5, 5, 4, 3, 3, 2, 2, 2	-20, -20, -20, -20, -20, -20, -20, -20	-160

Table 5.4: GA output from home.

Device	$P_{tx}(dBm)$	Total
5, 5, 4, 3, 3, 2, 2, 2	-20, -19, -20, -19, -18, -18, -16, -19	-149

The Genetic Algorithm output for the lab location, as shown in table 5.3, achieved the lowest possible transmission power of -20 dBm for all nodes in the network. This outcome is expected, given the network's short distances within the small lab setting. When devices are in close proximity to each other, there is no need to increase transmission power, as doing so would waste energy. In this scenario, GA successfully minimized transmission power for all nodes. Although it may appear that GA could have set the power even lower, it's important to note that -20 dBm is the lowest limit, and going below that would depend on the mathematical constraints covering path loss and RSSI sensitivity.

Conversely, table 5.4 displays the GA output for the home location, where transmission power was not set to the lowest possible value for all nodes due to the larger area. The GA output produced a transmission power range from -16 to -20 dBm, which is still an impressive result compared to the maximum transmission power mode. This variation in transmission power reflects the differing distances between devices within the home network.

Lastly, the device columns display the same types of devices, which is due to the total number of devices being set to 8, as specified in the parameters table 4.1. According to the mathematical constraints, this represents the optimal network configuration derived from the MCM output. If the total number of devices in the network were to be increased, the network configuration would exhibit greater variation.

In addition to these results, examining the plots for both the lab and home locations provides further insight into the transmission power optimization process based on the number of generations. The X-axis of the plots represents the total number of populations, with 100 max populations being set for this research, as mentioned in the parameters table 4.2. The maximum number of populations is adjusted depending on the optimization process and requirements. The linear curve observed in the plots is influenced by the parameters used in the GA process, such as distances between devices and the selection method. As these factors change, the transmission process is affected, resulting in different curve patterns in the plots. This demonstrates the flexibility of the GA in adapting to various network configurations and optimization objectives.

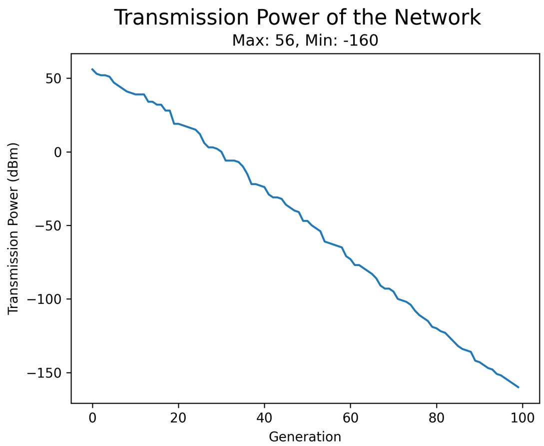

Figure 5.1: GA transmission power optimization for lab.

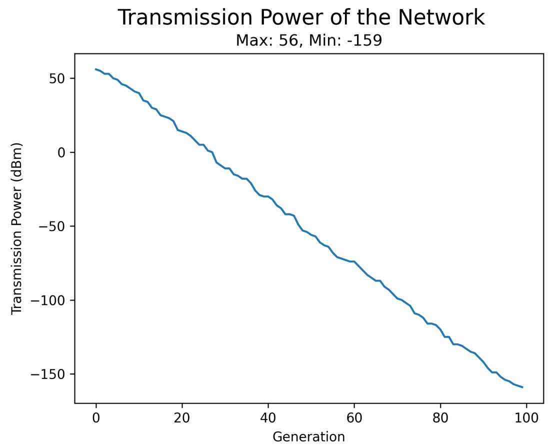

Figure 5.2: GA transmission power optimization for home.

5.3 Distance vs. Transmission Power Analysis

Understanding the relationship between distance and transmission power in the network is important for analyzing network configurations. By looking at the plots for both lab and home locations, this relationship becomes clearer. In these plots, the distance between devices is shown on the x-axis, while transmission power is displayed on the y-axis.

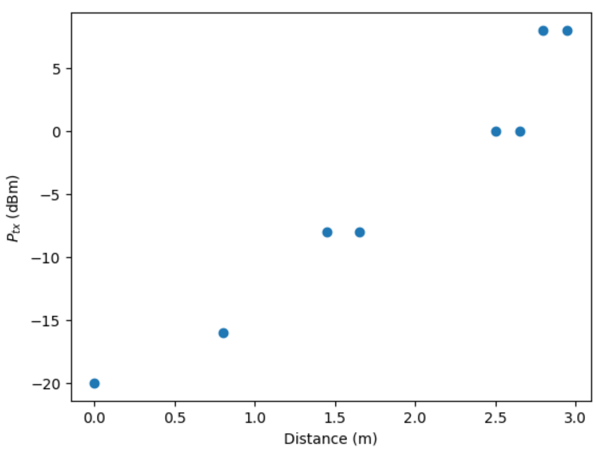

Figure 5.3: Transmission power vs. distance for lab location using MCM.

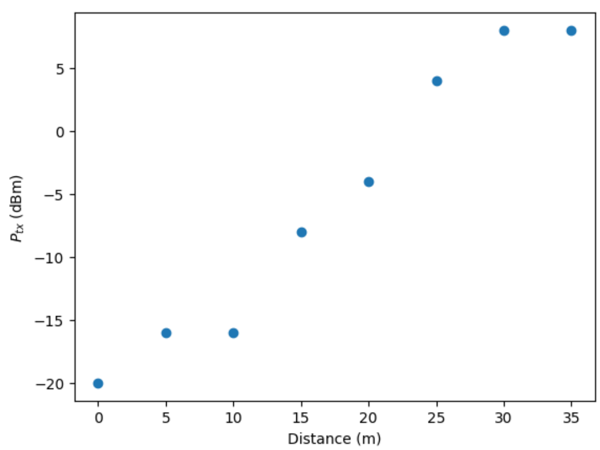

Figure 5.4: Transmission power vs. distance for home location using MCM.

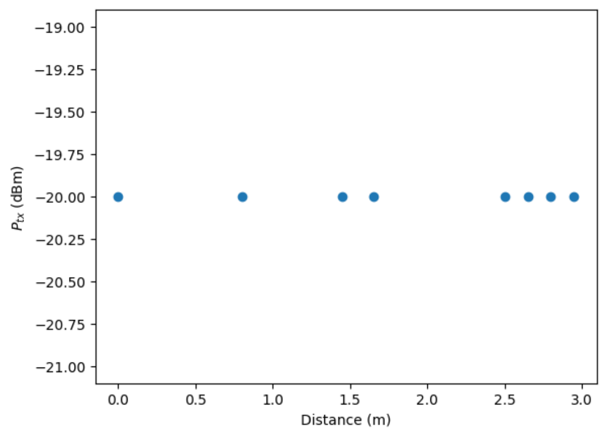

Figure 5.5: Transmission power vs. distance for lab location using GA.

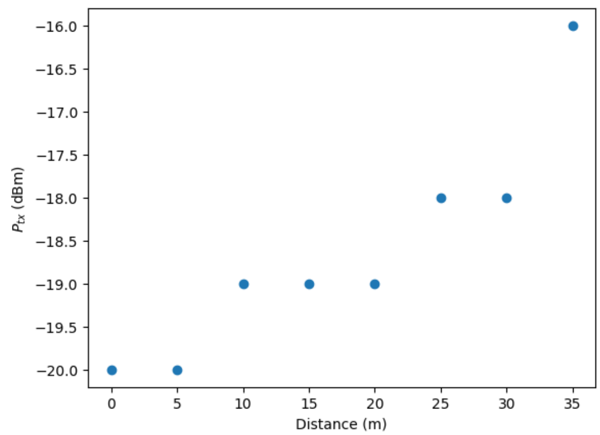

Figure 5.6: Transmission power vs. distance for home location using GA.

A closer look reveals that as the distance between devices increases, transmission power also increases. Conversely, devices closer to each other have lower transmission power settings. This pattern is expected because it is inefficient to use extra power for devices that are near each other. Instead, higher transmission power is needed to keep devices connected when they are farther apart.

Transmission power plays a key role in determining the coverage of Thread radio networks. Networks with higher transmission power settings can cover larger areas, ensuring that devices stay connected even when they are separated by greater distances [8]. Effective power management is important for optimizing network performance and saving energy.

5.4 Path Loss Analysis

The relationship between path loss, distance, and environment is a critical aspect of wireless communication. Two plots are provided to illustrate the path loss between devices at both lab and home locations. As anticipated, these plots demonstrate that the greater the distance between devices, the higher the path loss. It is important to note that even at close distances, higher path loss can occur due to environmental factors, as shown in table 2.1's path loss exponent.

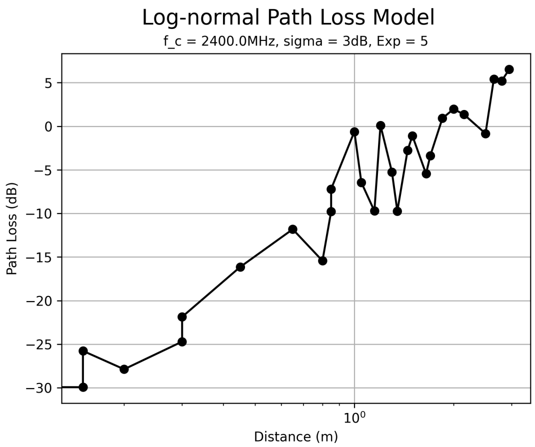

Figure 5.7: Path loss for lab location.

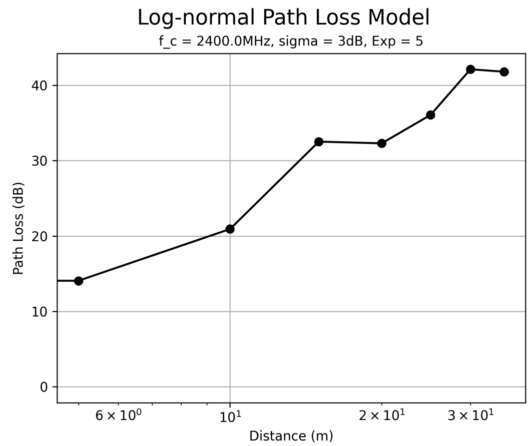

Figure 5.8: Path loss for hoome location.

Free space environments, such as satellite communication, typically exhibit lower path loss, while indoor locations like homes tend to experience higher path loss due to the presence of obstacles, such as walls and furniture [14]. The plots confirm the expected relationship between path loss, distance, and environment.

The first plot, representing the lab location, displays lower path loss between devices. This can be attributed to the controlled environment and shorter distances between devices. In contrast, the second plot, showcasing the home location, reveals higher path loss between devices. This increase in path loss can be attributed to both the larger distances between devices and the presence of obstacles in the home environment, which impede wireless communication signals.

The analysis of the path loss plots from the lab and home locations directly relates to the log-normal shadowing model, as shown in equation 2.7. By examining these real-world scenarios, the research highlights the importance of considering path loss and reinforces the applicability of the model in the power optimization process.

5.5 Algorithmic Parameter Analysis

So far, the analyses for MCM and GA have been based on the applied values to the experimental prototype, with the only differing parameter being the distance between the two different locations, lab and home. The previous experiments provided valuable insights into the performance of both algorithms within the tested parameters. However, it is also important to explore their behavior with different parameters and larger distances for a more comprehensive understanding. To achieve this, a parameter table has been created using the following imaginary values:

Table 5.5: Algorithmic parameter analysis.

Parameter	Value
d	10 m
D_0	0.35 m
n	6.0
σ	5.0 dB
Population size	30
Max iteration	20
Mutation rate	0.3
Selection method	Tournament
Mutation method	Random

These parameters are described in tables 4.1 and 4.2. Although increasing the distance would impact the number of devices and the computational power required for the MCM, the chosen distance represents a reasonable compromise for the given number of devices. In response to the sub-research question 3, the tables below present the MCM and GA outputs based on these different parameters, demonstrating how the algorithms behave under different conditions and larger distances:

Table 5.6: MCM output based on different parameters.

Device	$P_{tx}(dBm)$	Penalty
3, 4, 5, 5, 3, 2, 2, 2	-12, -12, 8, 4, -4, -16, -4, 8	0

Table 5.7: GA output based on different parameters.

Device	$P_{tx}(dBm)$	Total
5, 5, 4, 3, 3, 2, 2, 2	-11, -12, -9, 8, 6, -4, 5, 7	-10

While the MCM table may not show any significant differences compared to previous analyses, the transmission power from the GA table does present interesting observations. Upon closer inspection, the transmission power is no longer at the edge range, as seen in past analyses. This outcome is expected since the current transmission power is calculated from much larger distances, while the past analyses were based on closer distances. In smaller distances, lower transmission power values are sufficient to maintain a reliable connection between devices, as the signal propagation is stronger. On the other hand, larger distances require higher transmission power values to ensure the signal can effectively reach and maintain a stable connection with other devices in the network.

In the context of the GA plot for transmission power optimization, the optimization line is no longer linear, likely due to the different parameters used, especially the selection method. The max value reaches a high of 5901 dBm but drops to -17 dBm at the lowest point, a notable result. The Y-axis, which shows the transmission power made up of both penalties for each constraint violation and the output transmission power for each iteration from GA, is understandably higher. Even though the plot reached -17 dBm, the best transmission power is the one with no penalty, as shown by the GA output in the table.

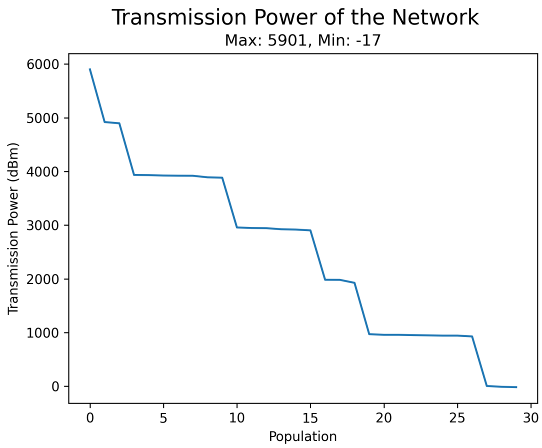

Figure 5.9: GA transmission power optimization based on different parameters.

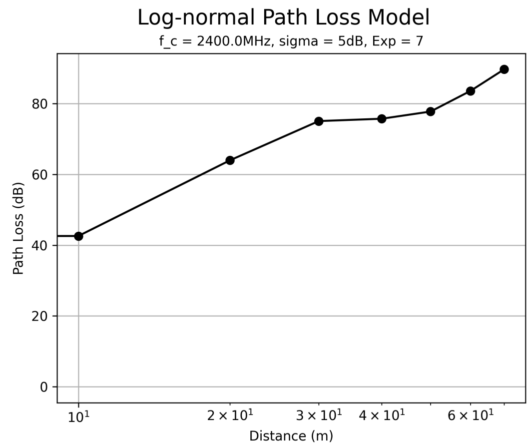

Figure 5.10: Path loss based on different parameters for large distance.

In addition to the impact on transmission power, the different parameters also affect path loss. The path loss plot, based on these different network parameters, shows a much higher path loss value as expected. This higher path loss is in line with the trend observed in past analyses, where path loss increased with higher distances. While distance is the primary contributor to these changes, other factors, such as the variance of components σ, reference distance d, and the signal power decay with distance in the path loss model n in equation 2.6, also play a role. The path loss figure 5.10 demonstrates these differences, providing valuable insights into how the various parameters influence path loss in different scenarios.

5.6 Experimental Data Analysis

The following analysis delves into a comprehensive data table that compares two distinct modes of operation in communication systems. The primary focus of this data table is to evaluate the current consumption of each mode, with the ultimate goal of identifying the more efficient method for power conservation. To accomplish this, an array of different parameters are considered. The subsequent sections provide an in-depth examination and interpretation of the data, aiming to answer the research questions and offer valuable insights into power optimization strategies.

Table 5.8: Experimental data analysis across different scenarios.

Method	Loc	Type	Mode	Ping	T (s)	P_{tx} (dBm)	Node	Current consumption (mA) I_1 Mean	I_1 Max	I_1 Min	I_2 Mean	I_2 Max	I_2 Min
Maximum	Lab	No Sensor	N/A	0	60	8	BR1	6.22	18.16	1.5	6.2	16.86	1.01
							BR2	6.43	18.83	1.48	6.43	17.77	1.12
							R1	9.72	18.83	7.3	9.78	21.22	7.44
							R2	9.73	19.11	7.12	9.73	20.53	7.35
							R3	9.65	18.12	7.16	9.78	20.14	7.53
							ED1	11.88	21.49	6.05	11.79	20.92	5.99
							ED2	11.87	21.39	6.14	11.69	21.07	6.13
							ED3	11.6	21.58	5.87	11.65	21.26	5.95
		Ping		50			BR1	6.29	17.96	1.51	6.27	18.16	1.28
							BR2	6.64	20.73	1.4	6.48	19.07	1.47
							R1	9.91	20.0	2.08	9.83	21.56	6.9
							R2	9.82	19.27	6.62	9.75	20.87	6.76
							R3	9.89	19.95	6.9	9.82	20.68	6.8
							ED1	11.86	21.66	6.08	11.83	22.35	6.08
							ED2	11.98	22.3	6.13	11.17	21.36	6.13
							ED3	11.89	21.9	6.04	11.66	21.8	5.95
	Home	No Sensor		0	300		BR1	6.62	18.2	1.01	6.2	17.05	1.07
							BR2	6.41	18.35	1.05	6.42	17.96	1.04
							R1	9.7	19.56	2.04	9.76	21.51	4.61
							R2	9.65	19.12	2.28	9.72	20.82	4.56
							R3	9.7	19.32	4.61	9.78	20.24	2.05
							ED1	11.73	21.26	5.99	11.76	21.17	6.04
							ED2	11.63	21.12	5.9	11.68	20.87	5.99
							ED3	11.73	21.46	5.81	11.64	21.61	5.81
		Ping		290			BR1	6.41	19.56	0.3	6.28	18.4	1.09
							BR2	6.62	20.39	1.08	6.49	19.32	1.05
							R1	9.92	20.14	4.56	9.85	21.66	2.33
							R2	9.83	19.61	4.65	9.76	21.07	4.65
							R3	9.89	19.85	2.14	9.84	21.87	2.04
							ED1	11.9	21.85	5.99	11.84	22.64	5.99
							ED2	11.77	21.61	5.99	11.75	21.46	5.99
							ED3	11.84	22.15	5.99	11.67	22.0	5.86
Optimized	Lab	No Sensor	MCM	0	60	-8	BR1	6.22	16.67	1.34	6.19	16.76	1.68
						8	BR2	6.45	18.93	1.66	6.41	18.35	1.78
						-16	R1	9.82	12.88	7.44	9.76	12.74	7.03
						0	R2	9.71	12.83	7.39	9.7	12.83	7.35
						0	R3	9.8	12.83	7.39	9.8	20.09	7.48
						-8	ED1	11.86	15.1	6.17	11.76	15.1	6.08
						-20	ED2	11.76	14.91	6.13	11.69	14.95	6.08
						8	ED3	11.78	21.17	6.13	11.64	21.66	5.99
			GA			-20	BR1	6.2	16.79	1.57	6.2	16.82	1.43
							BR2	6.42	15.88	1.6	6.41	17.17	1.57
							R1	9.77	12.88	7.3	9.76	12.83	7.39
							R2	9.68	12.83	7.12	9.73	12.83	7.44
							R3	9.76	12.6	7.12	9.78	12.79	7.21
							ED1	11.84	15.1	6.13	11.76	15.24	5.9
							ED2	11.7	14.81	6.08	11.64	15.0	6.04
							ED3	11.73	15.0	6.04	11.7	15.05	6.04
		Ping	MCM	50		-8	BR1	6.24	17.32	0.94	6.22	17.32	1.31
						8	BR2	6.6	20.09	1.28	6.48	19.17	1.38
						-16	R1	9.78	13.4	4.78	9.8	13.07	6.9
						0	R2	9.76	13.02	2.83	9.69	13.07	6.9
						0	R3	9.82	17.19	2.95	9.85	20.68	6.9
						-8	ED1	11.82	15.38	6.13	11.8	15.28	6.04
						-20	ED2	11.65	15.28	5.99	11.6	15.0	6.04
						8	ED3	11.88	21.76	6.04	11.66	21.76	5.99
			GA			-20	BR1	6.22	17.02	1.0	6.21	16.9	1.01
							BR2	6.43	17.18	1.28	6.42	16.92	1.05
							R1	9.76	13.3	4.78	9.79	13.21	7.03
							R2	9.71	13.02	4.69	9.69	13.07	6.85
							R3	9.79	13.21	5.01	9.77	13.02	6.9
							ED1	11.81	15.28	6.04	11.77	15.28	5.99
							ED2	11.7	15.19	6.13	11.6	15.0	5.99
							ED3	11.72	15.24	5.99	11.68	15.24	5.99
	Home	No Sensor	MCM	0	300	8	BR1	6.22	17.58	1.04	6.2	17.29	0.98
						8	BR2	6.42	18.16	1.06	6.41	18.4	1.12
						-8	R1	9.75	13.07	2.04	9.75	12.88	4.74
						-20	R2	9.73	12.93	4.74	9.71	12.83	4.69
						-4	R3	9.75	12.83	4.65	9.79	16.3	2.95
						-16	ED1	11.75	15.1	5.95	11.75	15.1	5.99
						4	ED2	9.73	12.93	4.74	9.71	12.83	4.69
						-16	ED3	9.75	12.83	4.65	9.79	16.3	2.95
			GA			-20	BR1	6.19	16.82	1.02	6.2	17.02	1.02
						-19	BR2	6.41	16.92	0.91	6.41	17.05	0.97
						-19	R1	9.76	12.83	1.99	9.75	12.93	4.78
						-20	R2	9.72	12.74	2.34	9.7	12.93	2.26
						-18	R3	9.75	12.88	4.74	9.78	12.88	4.69
						-18	ED1	11.75	15.1	5.9	11.76	15.1	5.99
						-16	ED2	11.67	14.95	5.99	11.63	15.24	5.99
						-19	ED3	11.73	15.1	6.08	11.68	18.98	0.25
		Ping	MCM	290		8	BR1	6.42	19.17	0.33	6.28	18.25	1.04
						8	BR2	6.63	19.51	1.05	6.49	19.32	1.13
						-8	R1	9.73	16.45	2.04	9.8	13.21	2.29
						-20	R2	9.67	13.21	2.02	9.69	13.02	4.65
						-4	R3	9.73	16.17	2.12	9.8	13.21	4.74
						-16	ED1	11.72	15.48	5.95	11.72	15.33	5.95
						4	ED2	11.69	18.11	5.9	11.65	18.25	5.95
						-16	ED3	11.65	15.28	5.95	11.67	15.24	5.86
			GA			-20	BR1	6.2	17.31	1.03	6.21	16.99	1.05
						-19	BR2	6.4	17.34	1.08	6.43	16.98	0.94
						-19	R1	9.73	13.76	2.01	9.79	13.3	2.28
						-20	R2	9.68	14.25	2.03	9.69	15.24	2.21
						-18	R3	9.76	13.21	4.56	9.78	13.12	4.65
						-18	ED1	11.74	15.28	5.99	11.78	15.33	6.08
						-16	ED2	11.64	15.24	5.99	11.62	15.19	5.95
						-19	ED3	11.69	15.52	5.9	11.66	15.28	5.9

In the analysis of the table 5.6, a detailed comparison between the maximum and optimized modes can be made, taking into account various parameters, including the mean, max, and min current values, location, iteration, and device type. Here is a more comprehensive overview of the data:

5.6.1 Mean, Max, and Min Current Analysis

The analysis of the mean, max, and min Current (mA) across all devices, locations, and methods revealed various trends. Overall, the mean current values ranged from 6.19 mA to 11.98 mA, with the lowest values observed in the Border Router (BR) series devices and the highest values in the End Device (ED) series devices. The maximum current values varied from 12.6 mA to 22.64 mA, while the minimum current values were between 0.25 mA and 7.53 mA. This broad range of values suggests that different devices, methods, and environments may have significant impacts on the current consumption of the devices tested.

5.6.2 Location-Specific Analysis

The location-specific analysis demonstrated that the devices' performance differed depending on whether they were tested in a lab or at home. In general, the mean, max, and min current values were higher in the lab setting compared to the home setting. This could be attributed to the controlled environment in the lab, which may have led to more stable and consistent performance across devices. This finding answer sub-research question 4, which aims to investigate the impact of location on power optimization performance.

5.6.3 Iteration-Specific Analysis

In comparing the first and second iterations, it was observed that the mean, max, and min current values showed little variation. This indicates that the performance of the devices was consistent across both iterations. However, some minor differences were noticed, such as a slight increase or decrease in the current values for some devices between iterations. This could be due to the variations in the environment or the devices' behavior during the testing period. This analysis answers sub-research question 4, which aims to explore the differences in power optimization performance between different iterations for both maximum and optimized modes.

5.6.4 Device-Specific Analysis

The device-specific analysis revealed that the border router series devices consistently exhibited the lowest mean, max, and min current values compared to the Router (R) and end device series devices. In contrast, the end device series devices had the highest mean, max, and min current values. This suggests that the border router series devices may be more energy-efficient than the other devices, while the end device series devices may require more power to operate. This finding answer sub-research question 4, which aims to compare the power optimization performance of different devices in maximum and optimized modes.

5.6.5 Type-Specific Analysis

When comparing devices with no sensor versus devices with a ping, it was found that devices without a sensor tended to have slightly lower mean, max, and min current values. This indicates that the presence of a sensor may increase power consumption in certain devices.

Further investigation into this finding revealed that devices with sensors require additional power to operate the sensor and transmit sensor data, leading to increased power consumption. In contrast, devices without sensors do not have these additional power requirements, resulting in lower power consumption overall.

5.6.6 Mode-Specific Analysis

The mode-specific analysis revealed that the devices' performance was affected by the MCM and GA modes. In general, the mean, max, and min current values were higher in the MCM mode compared to the GA mode. This suggests that the MCM mode require more power to operate and maintain, while the GA mode may offer more energy-efficient performance.

Furthermore, when comparing the power optimization performance of the MCM and GA modes, it was found that the GA mode outperformed the MCM mode in terms of energy efficiency. Devices operating in the GA mode consumed less power while still achieving comparable levels of performance, indicating that this mode may be a better option for power optimization. This finding supports the sub-research question 4 on the effect of mode on power optimization, indicating that the choice of mode can have a significant impact on power consumption and optimization performance.

5.6.7 Method-Specific Analysis

Lastly, the method-specific analysis showed that the mean, max, and min current values were lower in the optimized method compared to the maximum method. This indicates that the optimized method may provide a more energy-efficient solution for the devices tested, as it consumes less power overall. This insight could prove useful when selecting a method for future deployments to reduce energy consumption and improve device performance. This finding answer sub-research question 4, which aims to explore the correlation between mean, max, and min current values and the efficiency of power optimization for different methods.

In conclusion, the data analysis of the mean, max, and min current values across various parameters, including iteration, device type, location, sensor presence, mode, and method, revealed distinct trends in the devices' power consumption. The border router series devices consistently showed the lowest current values, suggesting better energy efficiency compared to other device types. Devices tested in the lab displayed higher current values than those in the home setting, indicating the influence of environmental factors on device power consumption.

Furthermore, devices without sensors generally consumed less power, and the GA mode demonstrated lower current values compared to the MCM mode. Finally, the optimized method appeared to be a more energy-efficient solution compared to the maximum method.

5.7 Experimental Results

In this section, the power efficiency of the devices under study is examined, focusing on the maximum and optimized methods applied to MCM and GA modes. The analysis considers the error values obtained from the first and second iteration MCM and GA values, providing insights into the variability and precision of the measurements. This results-driven perspective allows for a comprehensive understanding of the performance differences between the methods and modes under investigation.

Table 5.9: Experimental results across different scenarios with errors.

Method	Location	Type	Iteration (%)				Error (%)	
			I_1		I_2			
			Mode					
			MCM	GA	MCM	GA	MCM	GA
Optimized	Lab	No Sensor	25.69	26.42	20.6	26.31	5.09	0.11
		Ping	18.52	27.07	18.39	28.47	0.13	1.4
	Home	No Sensor	27.12	25.92	24.38	24.25	2.74	1.67
		Ping	19.24	26.19	24.84	27.47	5.6	1.28

The table 5.9 presents a comprehensive comparison of MCM and GA modes in the optimized method, with a focus on the percentage values calculated from the maximum current values obtained in previous analyses. Considering the maximum current values for the power optimization process is important because it helps identify the devices' peak current usage. This method offers a clearer understanding of the devices' power efficiency in different modes and iterations. By focusing on the highest current values, a more accurate assessment of the effectiveness of the power optimization process can be achieved, especially during the most demanding situations. This approach addresses the sub-research question 5, which aims to understand the impact of power optimization methods on devices' power efficiency across different modes and iterations.

In the optimized method, the lab location exhibits a higher percentage for no sensor and ping types in both MCM and GA modes when compared to the home location. Specifically, for the no sensor type, the lab location has a 25.69% and 26.42% improvement in MCM and GA modes, respectively, in the first iteration, while for the ping type, the lab location has an 18.52% and 27.07% improvement in MCM and GA modes, respectively. In the second iteration, the lab location maintains its higher performance with 20.6% and 26.31% improvements in MCM and GA modes for the no sensor type, and 18.39% and 28.47% improvements in MCM and GA modes for the ping type. This observation indicates that the devices in the lab location demonstrate better power efficiency.

On the other hand, in the home location, the no sensor type shows a 27.12% and 25.92% improvement in MCM and GA modes, respectively, in the first iteration, while for the ping type, there is a 19.24% and 26.19% improvement in MCM and GA modes, respectively. In the second iteration, the home location has a 24.38% and 24.25% improvement in MCM and GA modes for the no sensor type, and 24.84% and 27.47% improvements in MCM and GA modes for the ping type.

Errors were calculated based on the differences between the first and second iteration values for MCM and GA modes. The presence of errors might be attributed to various factors, such as device inconsistencies, environmental factors, or potential limitations in the experimental setup. These errors affect the research by introducing a level of uncertainty in the results, making it necessary to interpret the findings with caution, which addresses the second sub-research question 5. For instance, in the lab location, the no sensor type has errors of 5.09% and 0.11% in MCM and GA modes, while in the home location, errors are 2.74% and 1.67% for the same modes.

In conclusion, the analysis demonstrates the effectiveness of parameter optimization in developing a power-optimized Thread mesh wireless network, addressing the main research question and the problem definition. Both MCM and GA modes outperform the maximum method, with GA optimization consistently offering better optimization results than MCM across different locations and device types. This indicates that the GA approach significantly contributes to lowering power consumption in Thread mesh wireless networks by optimizing transmission power parameter more effectively than the MCM method. The findings of this study align with those of Girgis, Mahmoud, Abdullatif, *et al.* [22], who found GA to be effective in minimizing network costs in small-size networks, and Sheth and Han [8], who achieved a maximum power savings of 25% in 802.11b wireless networks by implementing an adaptive transmit power control algorithm.

The algorithmic approach, specifically the GA optimization, can be integrated into the system by adjusting transmission power parameter according to the optimization results. By monitoring the network conditions and transmission power, the Thread mesh wireless network can maintain optimal energy efficiency. The results provide a solid foundation for future exploration and enhancements in power optimization using algorithmic approaches, addressing the challenges of consuming higher power, ultimately realizing the full potential of Thread-based wireless communication in a wide range of low-powered IoT network fields.

Chapter 6

Conclusions and Recommendations

6.1 Conclusions

This research on power optimization in Thread mesh wireless networks using transmission power as a parameter has demonstrated the effectiveness of algorithmic approaches, particularly Genetic Algorithm, in reducing power consumption. GA optimization consistently outperformed both Monte Carlo Method mode and maximum method across different locations and device types, with improvements of up to 28.47% in power efficiency and error rates as low as 0.11%. MCM also achieved improvements of up to 27.12% in power efficiency, while errors reached up to 5.6%. These results not only enhance the performance of MOOD-Sense initiatives and other IoT applications but also contribute to sustainable and energy-efficient IoT network implementation. By adhering to responsible research and innovation principles, this study ensures the development of an optimized system design adaptable for various applications beyond MOOD-Sense, promoting energy-conserving, environmentally friendly, and sustainable IoT devices and network integration. This research demonstrates that optimizing transmission power using algorithmic approaches, specifically GA optimization, can significantly reduce power consumption in Thread mesh wireless networks, paving the way for future exploration and enhancements in power optimization using algorithmic approaches, addressing the challenges of consuming higher power, and ultimately realizing the full potential of Thread-based wireless communication in a wide range of low-powered fields.

6.2 Recommendations

Considering the conclusions from this research, several recommendations for future work are proposed to further enhance power optimization in Thread mesh wireless networks. These suggestions aim to build on the foundation laid by this research and contribute to the ongoing development of Thread mesh wireless networking technologies.

1. **Dynamic Transmission Power Allocation**: Develop a custom SDK on top of existing platforms like Zephyr, nRF, or OpenThread that automatically sets the

transmission power based on the distances between devices without requiring manual action and reflashing the device. By automating this process, the network can achieve better energy efficiency, adapt to changes in device locations more effectively, and minimize the need for human intervention to update transmission power settings, making the Thread network more sustainable and user-friendly.

2. **Exploring Different Thread Devices**: Investigate the impact of different Thread devices, such as Full Thread Device (FTD), Minimal Thread Device (MTD), and Sleepy Thread Device (STD), on power consumption. By understanding the unique characteristics and energy requirements of each device type, the most suitable Thread devices can be selected to improve overall network efficiency. A thorough evaluation of device capabilities, power requirements, and application-specific needs can help guide the selection process for an optimized network configuration.

3. **Investigating Low-Power SoC Options**: Assess various low-power SoC options available on the market to determine the most energy-efficient solutions for the Thread network. By considering different devices with better low-powered SoC capabilities, the overall energy consumption of the network can be reduced, leading to a more sustainable and efficient network. This exploration can help identify devices that meet the performance requirements of the network while minimizing power consumption and maximizing energy efficiency.

Implementing these recommendations can help future research advance the optimization of Thread mesh wireless networks, ultimately leading to more efficient IoT wireless networking solutions.

Definitions and Abbreviations

ANT Adaptive Network Topology.

ARM Advanced RISC Machine.

BLE Bluetooth Low Energy.

DK Development Kit.

FTD Full Thread Device.

GA Genetic Algorithm.

ICMP Internet Control Message Protocol.

IEEE Institute of Electrical and Electronics Engineers.

IoT Internet of Things.

IP Internet Protocol.

IPv6 Internet Protocol version 6.

LOS Line of Sight.

MAC Media Access Control.

MCM Monte Carlo Method.

MTD Minimal Thread Device.

nRF Nordic Radio Frequency.

PC Personal Computer.

PPK Power Profiler Kit.

RCP Radio Coprocessor.

REED Router Eligible End Device.

RSSI Received Signal Strength Indicator.

SA Simulated Annealing.

SDK Software Development Kit.

SED Sleepy End Device.

SoC System on Chip.

STD Sleepy Thread Device.

TDMA Time Division Multiple Access.

USB Universal Serial Bus.

UWB Ultra-Wide Band.

WMN Wireless Mesh Network.

WSN Wireless Sensor Network.

References

[1] H. Groningen, *Sensor project helps people with dementia and their nurses | hanze uas*, en. [Online]. Available: https://www.hanze.nl/en/research/centres/centre-of-expertise-healthy-ageing/stories/sensor-project-helps-people-with-dementia-and-their-nurses (visited on 03/31/2023).

[2] T. Group, *Thread Benefits*, en. [Online]. Available: https://www.threadgroup.org/What-is-Thread/Thread-Benefits (visited on 03/31/2023).

[3] T. Group, *Thread network fundamentals*, en, May 2020. [Online]. Available: https://www.threadgroup.org/Portals/0/documents/support/Thread%20Network%20Fundamentals_v3.pdf.

[4] N. Semiconductor, *Nrf52840 dk product brief*, en, version 2.0. [Online]. Available: https://www.nordicsemi.com/-/media/Software-and-other-downloads/Product-Briefs/nRF52840-DK-product-brief.pdf.

[5] N. Semiconductor, *Battery life estimation for thread and zigbee seds*, en, The goal is to provide a simple mathematical model based on measurements to determine the battery life for any application in a few representative scenarios., Oct. 2021. [Online]. Available: https://infocenter.nordicsemi.com/pdf/nwp_039.pdf.

[6] D. P. Kroese, T. Brereton, T. Taimre, and Z. I. Botev, "Why the monte carlo method is so important today," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 6, no. 6, pp. 386–392, 2014.

[7] A. Lambora, K. Gupta, and K. Chopra, "Genetic algorithm-a literature review," in *2019 international conference on machine learning, big data, cloud and parallel computing (COMITCon)*, IEEE, 2019, pp. 380–384.

[8] A. Sheth and R. Han, "An implementation of transmit power control in 802.11 b wireless networks," *Department of Computer Science University of Colorado, CU-CS-934-02*, pp. 3–6, 2002.

[9] Z. Y. Wu, H. Gong, S. J. Li, M. H. Yu, and S. Y. Wang, "Study of the factors affecting wireless transmission on jn5148," in *Advanced Materials Research*, Trans Tech Publ, vol. 926, 2014, pp. 2110–2113.

[10] I. Dokmanic, R. Parhizkar, J. Ranieri, and M. Vetterli, "Euclidean distance matrices: Essential theory, algorithms, and applications," *IEEE Signal Processing Magazine*, vol. 32, no. 6, pp. 12–30, 2015.

[11] K. Benkic, M. Malajner, P. Planinsic, and Z. Cucej, "Using rssi value for distance estimation in wireless sensor networks based on zigbee," in *2008 15th international conference on systems, signals and image processing*, IEEE, 2008, pp. 303–306.

[12] F. Shang, W. Su, Q. Wang, H. Gao, and Q. Fu, "A location estimation algorithm based on rssi vector similarity degree," *International Journal of Distributed Sensor Networks*, vol. 10, no. 8, p. 371 350, 2014. DOI: 10.1155/2014/371350. eprint: https://doi.org/10.1155/2014/371350. [Online]. Available: https://doi.org/10.1155/2014/371350.

[13] N. Semiconductor, *Nrf52840 product specification*, en, version 1.0, 2018. [Online]. Available: https://infocenter.nordicsemi.com/pdf/nRF52840_PS_v1.0.pdf.

[14] Y. S. Cho, J. Kim, W. Y. Yang, and C. G. Kang, *MIMO-OFDM wireless communications with MATLAB*. John Wiley & Sons, 2010.

[15] A. Norouzi and A. H. Zaim, "Genetic algorithm application in optimization of wireless sensor networks," *The Scientific World Journal*, vol. 2014, 2014.

[16] N. Semiconductor, *Nrf52840 dongle product brief*, en, version 2.0. [Online]. Available: https://www.nordicsemi.com/-/media/Software-and-other-downloads/Product-Briefs/nRF52840-Dongle-product-brief.pdf.

[17] N. Semiconductor, *Power profiler kit ii product brief*, en, version 1.0. [Online]. Available: https://nsscprodmedia.blob.core.windows.net/prod/software-and-other-downloads/product-briefs/power-profiler-kit-ii-pbv10.pdf.

[18] A. Alm, *Internet of things mesh network: Using the thread networking protocol*, 2019.

[19] T. Group, *The value of low power*, Feb. 2018. [Online]. Available: https://www.threadgroup.org/Portals/0/documents/support/TheValueofLowPowerWhitepaper_2454_2.pdf.

[20] E. Azoidou, Z. Pang, Y. Liu, D. Lan, G. Bag, and S. Gong, "Battery lifetime modeling and validation of wireless building automation devices in thread," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 7, pp. 2869–2880, 2017.

[21] A. Behzad and I. Rubin, "High transmission power increases the capacity of ad hoc wireless networks," *IEEE Transactions on Wireless Communications*, vol. 5, no. 1, pp. 156–165, 2006. DOI: 10.1109/TWC.2006.1576539.

[22] M. R. Girgis, T. M. Mahmoud, B. A. Abdullatif, and A. M. Rabie, "Solving the wireless mesh network design problem using genetic algorithm and simulated annealing optimization methods," *International Journal of Computer Applications*, vol. 96, no. 11, 2014.

[23] K. P. Ferentinos, T. A. Tsiligiridis, and K. G. Arvanitis, "Energy optimization of wireless sensor networks for environmental measurements," in *Proceedings of the International Conference on Computational Intelligence for Measurment Systems and Applicatons (CIMSA)*, vol. 51, 2005, pp. 1031–1051.

Appendix

This Appendix provides various resources and links related to the research project. These resources include the full dataset analysis, algorithms, custom implementations, and output datasets. Due to these materials' large size and complexity, it is not feasible to include them directly in the research paper. Instead, the links in the following sections grant access to the complete datasets, algorithms, and implementations, allowing interested readers to explore the project in greater detail and better understand the methodology, optimization techniques, and findings. The sections below outline the resources available in the appendix.

1 Dataset Analysis

This section provides the link to the dataset analysis repository on GitLab. This repository contains a comprehensive set of analyses performed for the project. Due to the extensive nature of the analyses, including them all in this paper is not feasible. By sharing the repository, readers can access detailed studies and better understand the project's intricacies. The repository can be accessed using the following link: `https://gitlab.com/mmikhan/threadpowerprofiler/`

2 Dataset

The complete dataset, too large to include within the research paper, is available on GitLab. This dataset contains detailed information on the performance of the Thread network under various conditions and configurations. The original dataset is in binary format but has been converted to CSV for convenience and easier access. Access the dataset here: `https://gitlab.com/mmikhan/threadpowerprofiler/`

3 Algorithm

This section links the complete algorithm consisting of the MCM and GA implementations on GitHub. This repository houses the code and documentation required to understand and replicate the optimization techniques used in this research project. Access the algorithm here: `https://github.com/mmikhan/ThreadNetPowerOptGA`

4 nRF Thread Client and Server Custom Implementation

This part presents the custom implementation of the nRF Thread Client and Server used in the physical prototype. This implementation was essential to successfully deploying and testing the optimized Thread network. Access the nRF Thread Client and Server custom implementation here: `https://github.com/mmikhan/Connecta`

5 Optimization Results

Finally, this section provides access to the large output dataset from MCM and GA simulations. This dataset is crucial for understanding the outcomes of the optimization techniques and their impact on the energy efficiency and performance of the Thread network. Access the output dataset here: `https://gitlab.com/mmikhan/threadpowerprofiler/`

The resources presented in the appendix thoroughly examine the research project, its methodology, and the optimization techniques utilized. Through carefully studying these materials, a comprehensive understanding of the project's development, implementation, and outcomes can be obtained, thereby enriching the overall context of the research.